

Oracle® Fusion Middleware

System Administrator's Guide for Oracle Business Intelligence Enterprise Edition



12c (12.2.1.3.0)

E80930-05

October 2018

The Oracle logo, consisting of the word "ORACLE" in white, uppercase letters, centered within a solid red square.

ORACLE®

Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition, 12c (12.2.1.3.0)

E80930-05

Copyright © 2015, 2018, Oracle and/or its affiliates. All rights reserved.

Primary Author: Kate Price

Contributing Authors: Marla Azriel, Christine Jacobs, Nick Fry, Stefanie Rhone.

Contributors: Oracle Business Intelligence development, product management, and quality assurance teams.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience	xxii
Documentation Accessibility	xxii
Related Documentation and Other Resources	xxii
Conventions	xxiii

New Features for Oracle Business Intelligence System Administrators

New Features and Changes for Oracle BI EE 12 c (12.2.1.3.0)	xxiv
New Features and Changes for Oracle BI EE 12 c (12.2.1.2.0)	xxiv
New Features and Changes for Oracle BI EE 12c (12.2.1.1.0)	xxiv
New Features and Changes for Oracle BI EE 12c (12.2.1.0)	xxiv

Part I Administering Oracle Business Intelligence

1 Introduction to Oracle Business Intelligence System Administration

What Are the Oracle Business Intelligence System Administration Tasks?	1-1
Getting Started with Managing Oracle Business Intelligence	1-2
What Is the Oracle Business Intelligence System Logical Architecture?	1-3
Oracle Business Intelligence System Architecture	1-4
Oracle Business Intelligence Components	1-4
About the Administration Server, Managed Servers, and System Components	1-5
Key Directories in Oracle Business Intelligence	1-7
What System Administration Tools Manage Oracle Business Intelligence?	1-7
Fusion Middleware Control	1-8
Oracle WebLogic Server Administration Console	1-8
Process Control Commands	1-9
Oracle WebLogic Scripting Tool (WLST)	1-9
Oracle BI AdministrationTool	1-9
Catalog Manager	1-9

Job Manager	1-10
Working with the Sample Application	1-10
Oracle BI Publisher Integration	1-10
Topics of Interest in Other Guides	1-10
System Requirements and Certification	1-11

Part II Managing Processes and Components

2 Managing Oracle Business Intelligence Processes

About Managing Oracle Business Intelligence Processes	2-1
Conditions for Starting the Oracle Business Intelligence System	2-1
Using Commands to Start, Stop, and View Status of Oracle BI EE Processes	2-2
Starting Oracle Business Intelligence Component Processes in a Domain	2-2
Stopping Oracle Business Intelligence Component Processes in a Domain	2-3
Viewing the Status of Oracle Business Intelligence Components in a Domain	2-4
Using Fusion Middleware Control to Start and Stop BI System Component Processes	2-5
Using Fusion Middleware Control to Start and Stop Java Components	2-6
Using Oracle WebLogic Server Administration Console to Start and Stop Java Components	2-7

Part III Scaling and Deploying for High Availability and Performance

3 Scaling Your Deployment

About Scaling Oracle Business Intelligence	3-1
Setting Up Shared Files and Directories	3-3
Changing the Singleton Data Directory (SDD)	3-3
Setting Up the Global Cache	3-4
Managing Capacity in Oracle Business Intelligence (Vertically Scaling)	3-4
Adding System Components	3-5
Removing System Components	3-7
Managing Availability in Oracle Business Intelligence (Horizontally Scaling)	3-7
Adding New Computers	3-8
Removing Existing Computers	3-11
Validating That Your System Has Been Scaled Correctly	3-12
Using Fusion Middleware Control to View System Component Availability	3-12
Using the Administration Console to View Managed Server Availability	3-13

4 Deploying Oracle Business Intelligence for High Availability

About Oracle Business Intelligence Components in a Clustered Environment	4-1
Recommendations for Availability	4-3
Using Fusion Middleware Control to Identify Single Points of Failure	4-3
Achieving High Availability Using an Active-Passive Model	4-3
Configuring Oracle Business Intelligence Components for High Availability	4-4
Optional Configuration for Oracle Business Intelligence High Availability	4-4
Setting Optional Cluster Controller Parameters	4-4
Setting Optional Presentation Services Parameters	4-5
Setting Optional Oracle BI Presentation Services Plug-in Parameters	4-6
Using the Cluster Manager	4-6
Viewing and Managing Cluster Information	4-7
Status Information	4-7
Cache Information	4-8
Session Information	4-9
Server Information	4-10
Troubleshooting a Fusion Middleware Control Clustered Environment	4-11
Avoiding Errors with Network Appliance Devices When the Oracle BI Server Is Running on Linux or UNIX	4-11

5 Managing Performance Tuning and Query Caching

Monitoring Service Levels	5-1
Using Fusion Middleware Control to View All Oracle Business Intelligence Metrics	5-2
Using the Administration Console to View Metrics for Java Components	5-2
About Query Performance Tuning	5-3
Setting Performance Parameters in Fusion Middleware Control	5-4
Using Fusion Middleware Control to Disallow RPD Updates	5-4
Using Fusion Middleware Control to Set the User Session Log-Off Period	5-5
Using Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables	5-5
Using Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View	5-6
About the Oracle BI Server Query Cache	5-7
Query Cache Architecture	5-7
Advantages of Caching	5-8
Costs of Caching	5-8
Disk Space	5-9
Administrative Tasks	5-9
Keeping the Cache Up To Date	5-9
CPU Usage and Disk I/O	5-9

Cache Sharing Across Users	5-10
About the Refresh Interval for XML Data Sources	5-10
About the Global Cache	5-10
Configuring Query Caching	5-12
Using Fusion Middleware Control to Enable and Disable Query Caching	5-13
Using Fusion Middleware Control to Set Query Cache Parameters	5-13
Manually Editing Additional Query Cache Parameters	5-13
Using Fusion Middleware Control to Set Global Cache Parameters	5-14
Manually Editing Additional Global Cache Parameters	5-14
Monitoring and Managing the Cache	5-15
Choosing a Cache Management Strategy	5-15
Disabling Caching for the System	5-15
Caching and Cache Persistence Timing for Specified Physical Tables	5-15
Configuring Oracle BI Server Event Polling Tables	5-16
Purging and Maintaining Cache Using ODBC Procedures	5-16
About ODBC Procedure Syntax	5-17
About Sharing the Presentation Services Query Cache	5-18
About Result Records	5-18
Storing and Purging Cache for SAP/BW Data Sources	5-18
How Repository Changes Affect the Query Cache	5-20
Strategies for Using the Cache	5-21
About Cache Hits	5-21
Ensuring Correct Cache Results When Using Row-Level Database Security	5-24
Running a Suite of Queries to Populate the Cache	5-24
Using Agents to Seed the Oracle BI Server Cache	5-25
Using the Cache Manager	5-26
Displaying Global Cache Information in the Cache Manager	5-27
Purging Cache in the Administration Tool	5-28
Cache Event Processing with an Event Polling Table	5-29
Setting Up Event Polling Tables on the Physical Databases	5-30
Making the Event Polling Table Active	5-32
Populating the Oracle BI Server Event Polling Table	5-32
Troubleshooting Problems with Event Polling Tables	5-33
Configuring the Data Source Cache	5-33
Creating a Data Source Cache on the File System	5-33
Creating a Data Source Cache on a Database	5-35
Managing the Data Source Cache	5-37
Managing the Oracle BI Presentation Services Cache Settings	5-39
Improving Oracle BI Web Client Performance	5-40
Configuring Apache HTTP Server for Static File Caching	5-41
Configuring Oracle HTTP Server for Static File Caching	5-43

Setting the JVM Heap Size for Oracle Business Intelligence	5-44
Improving Performance when Downloading Large Reports	5-44
Capturing Metrics Using the Dynamic Monitoring Service	5-45
Using the Dynamic Monitoring Service for Metrics	5-45
Using WLST Commands for Metrics	5-46

Part IV Resolving Issues

6 Diagnosing and Resolving Issues in Oracle Business Intelligence

What Diagnostic Tools Are Available?	6-1
Collecting Diagnostic Bundles	6-2
Viewing and Configuring Diagnostic Log Files	6-2
Using Fusion Middleware Control to View Log Information, Error Messages, and Alerts	6-3
Configuring Log File Rotation Policy and Specifying Log Levels	6-4
Using Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels	6-4
Manually Changing Additional Log File Settings	6-5
Diagnosing Issues Using the Log Viewer	6-5
Understanding Diagnostic Log Files and Log Configuration Files	6-6
What Are Diagnostic Log Files and Where Are They Located?	6-7
What Are Diagnostic Log Configuration Files and Where Are They Located?	6-8
What Are Log File Message Categories and Levels?	6-10
What is Log File Rotation?	6-11
What Messages Are Included in the System Log?	6-12
Managing the Query Log	6-12
Configuring Query Logging	6-13
Setting the Query Logging Level	6-13
Setting the Query Logging Level for a User	6-15
Using the Log Viewer	6-15
Running the Log Viewer Utility	6-15
Interpreting the Log Records	6-16
Logging in Oracle BI Server	6-17
Using the Oracle BI Presentation Services Logging Facility	6-17
Setting the Logging Levels for Oracle BI Presentation Services	6-18
Structure for the Oracle BI Presentation Services Configuration File	6-18
Examples of the Formats of Logged Messages	6-22
Oracle BI Presentation Services Message Structure	6-23
Oracle BI Presentation Services Log Filters	6-23
Diagnosing Issues with Agents	6-24

Debugging Agents Using Fusion MiddlewareControl	6-24
Manually Debugging Agents Using instanceconfig.xml	6-25
Using ODBC/JDBC Procedures to Obtain Oracle BI Server Diagnostics	6-26
About the Oracle BI Server ODBC/JDBC Procedures	6-26
Obtaining a List of Available Diagnostic Categories	6-26
Running Specific Diagnostics	6-27
About Parameters for ODBC/JDBC Procedures	6-28
Troubleshooting System Startup	6-30
Administration Server Fails to Start When the Database Is Not Running	6-30
Managed Server Is Down	6-30
Oracle BI Server Fails to Start	6-31
Cannot Log In	6-31

7 Managing Usage Tracking

About Usage Tracking	7-1
Setting Up Direct Insertion to Collect Information for Usage Tracking	7-2
Setting Up the Usage Tracking Statistics Database	7-2
Setting Direct Insertion Parameters	7-2
Setting Optional Direct Insert Parameters	7-3
Description of the Usage Tracking Data	7-4

Part V Configuring Oracle Business Intelligence

8 Using Tools for Managing and Configuring Oracle Business Intelligence

Why Use Fusion Middleware Control and WebLogic Server Administration Console?	8-1
Managing Oracle Business Intelligence System Components Using Fusion Middleware Control	8-2
Logging into Fusion Middleware Control	8-2
Displaying Oracle Business Intelligence Pages in Fusion Middleware Control	8-3
Using the Navigation Tree in Fusion Middleware Control	8-4
Tips for Using Fusion Middleware Control with Oracle Business Intelligence	8-4
Configuring Oracle Business Intelligence System Settings	8-5
Using Fusion Middleware Control	8-6
Using a Text Editor	8-7
Using the WebLogic Scripting Tool (WLST)	8-7
Making Offline Configuration Changes	8-7
Making Online Configuration Changes	8-8

Updating the Java Development Kit (JDK)	8-9
Managing Oracle Business Intelligence Java Components Using the Oracle WebLogic Server Administration Console	8-9

9 Managing Metadata and Working with Service Instances

About Oracle Business Intelligence Application Archive (BAR) Files	9-1
What Are Oracle Business Intelligence Application Archive (BAR) Files?	9-1
What Predefined BAR Files are Available?	9-1
About Importing BAR Files	9-3
Managing Service Instances	9-3
listBIServiceInstances	9-5
getBIServiceInstance	9-5
scaleOutBIServiceInstance	9-5
export_service_instance	9-6
import_bar	9-7
refreshServiceInstance	9-8
refreshDomainServiceInstances	9-9
resetServiceInstance	9-9

10 Configuring Connections to External Systems

Configuring Email and Agents	10-1
Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents	10-1
Configuring for Actions with the Action Framework	10-2
Configuring Connections to the Marketing Content Server	10-2
Configuring Connections to Data Sources	10-3

11 Configuring Presentation Setting Defaults

Using Fusion Middleware Control to Change Presentation Setting Defaults	11-1
---	------

12 Configuring Mapping and Spatial Information

What Are the System Requirements for Map Views?	12-1
Hardware Sizing and Deployment Strategy for Maps	12-3
Sample Spatial Dataset for use with Map Views	12-3
Administering Maps	12-4
Working with Maps and Layers	12-4
Associating Layers with Columns	12-4
Ordering Layers on Maps	12-5

Changes to Spatial Metadata Require Restart	12-7
Administration Page Functions	12-7
Administering Maps Using Administration Pages	12-7
Handling the Translation of Layers in Maps	12-8

13 Configuring Time Zones

Why and Where Are Time Zones Used?	13-1
Setting Time Zones	13-2
What Is the Precedence Order for Time Zones?	13-3
User-Preferred Time Zone	13-3
Where Are Time Zone Specifications Stored?	13-3
Description of Time Zone Settings	13-4
Example: Configuration File Settings for Specifying the Time Zone	13-5

14 Localizing Oracle Business Intelligence

What Is Localization?	14-1
What Components Are Translated?	14-1
Tasks for Localizing Oracle Business Intelligence Components	14-2
Localizing Oracle BI PresentationServices	14-3
Localizing the User Interface for Oracle BI PresentationServices	14-3
Understanding the Directory Structure for Localizing Presentation Services	14-4
Localizing Messages for Users' Preferred Currency	14-4
Specifying the Default Language for the Sign-In Page	14-5
Configuring the Languages and Locales for the Sign-In Page	14-5
Specifying the Scaling of Numbers in Performance Tiles	14-6
Specifying the Language in the URL	14-7
Localizing Oracle BI Presentation Catalog Captions	14-8
Step 1: Understanding the Export Process	14-8
Step 2: Exporting Text Strings in the Catalog	14-9
Step 3: Editing Exported Strings in XML Files	14-10
Step 4: Handling Duplicate Exported Text Strings	14-11
Step 5: Exposing Text Strings in the Catalog	14-12
Tip for Arabic and Hebrew in Mozilla Firefox Browsers	14-13
Setting the Current Locale in Catalog Manager	14-14
Setting the Current Locale in the Oracle BI Server	14-14
Setting Locale Parameters on the Oracle BI Server	14-15
Setting the Locale on UNIX Systems	14-15
Understanding How the Error Message Language Is Determined	14-16
Setting the Language for Components of the Oracle BI Server	14-17

Modifying the Language of the User Interface for the Administration Tool	14-17
Troubleshooting the Current Locale in the Oracle BI Server	14-18
Handling the NLS Locale Not Supported Error Message	14-18
Setting the Japanese Locale on AIX Systems	14-18
Ensuring That Text for Oracle BI Server Utilities is Displayed in the Correct Language	14-19
Modifying the Metadata Repository When the Underlying Oracle Database NLS_CHARACTERSET Is Not Unicode	14-19
Localizing Metadata Names in the Repository	14-19
Supporting Multilingual Data	14-21
What is Multilingual Data Support?	14-22
What is Lookup?	14-22
What is Double Column Support?	14-22
Designing Translation Lookup Tables in a Multilingual Schema	14-22
A Lookup Table for Each Base Table	14-23
A Lookup Table for Each Translated Field	14-23
Creating Logical Lookup Tables and Logical Lookup Columns	14-24
Creating Logical Lookup Tables	14-24
Designating a Logical Table as a Lookup Table	14-25
About the LOOKUP Function Syntax	14-26
Creating Logical Lookup Columns	14-26
Creating Physical Lookup Tables and Physical Lookup Columns	14-28
Supporting Multilingual Data in Essbase Through Alias Tables	14-30
Enabling Lexicographical Sorting	14-30

15 Configuring Currency Options

Changing the Default Currency for Analyses	15-1
Defining User-Preferred Currency Options	15-2
Defining User-Preferred Currency Options Using a Static Mapping	15-4
Example: Static Mapping to Define User-Preferred Currency Options	15-5
Defining User-Preferred Currency Options Using a Dynamic Mapping	15-6
Example: Dynamic Mapping to Define User-Preferred Currency Options	15-7

16 Managing How Content is Indexed and Searched

Scheduling Regular Content Crawls of Catalog Objects	16-1
Scheduling Regular Content Crawls of Repository Content	16-2
Monitoring Search Crawl Jobs	16-2

17 Configuring and Managing the Oracle BI Presentation Catalog

About the Oracle BI Presentation Catalog	17-1
Objects in the Catalog	17-1
File System Guidelines for Catalogs	17-3
Maintaining the Oracle BI Presentation Catalog	17-5
Manually Changing Configuration Settings for the Catalog	17-5
Deploying Catalogs and Objects to Production	17-6
Deploying Catalogs to Production	17-6
Deploying Objects to Production	17-7
Updating Catalog Objects	17-7
About Catalog Manager	17-8
Starting Catalog Manager and Opening Catalogs	17-9
Requirements for Running Catalog Manager	17-9
Starting the Catalog Manager User Interface	17-10
Resolving Startup Issues on Linux Systems	17-10
Understanding the Two Catalog Modes	17-11
Operations Available in Online Mode and Offline Mode	17-12
Opening an Oracle BI Presentation Catalog	17-13
Using the Catalog Manager Workspace	17-14
What Does the Catalog Manager Workspace Do?	17-14
What Does the Catalog Manager Workspace Look Like?	17-14
Managing the View of the Catalog Manager Workspace	17-15
Working with Objects in Catalog Manager	17-15
Searching for Catalog Objects Using Catalog Manager	17-16
Copying and Pasting Objects	17-17
Tips for Copying and Pasting	17-17
Advanced Options for Pasting Objects	17-18
Renaming Catalog Objects	17-20
Working with the Properties of Catalog Objects	17-21
Setting Permissions of Catalog Objects	17-22
Previewing Objects from Catalog Manager	17-23
Validating the Catalog Using Catalog Manager	17-24
Viewing and Editing Catalog Objects in XML	17-25
Searching for and Replacing Catalog Text Using Catalog Manager	17-27
Searching for and Replacing a Simple Catalog Text String	17-27
About Searching for and Replacing Multiple Catalog Text Strings	17-27
XML File Format for Searching for and Replacing Text Strings	17-27
Example XML File for Searching for and Replacing Text Strings	17-28
Searching for and Replacing Multiple Catalog Text Strings	17-29
Creating Reports to Display Catalog Data Using Catalog Manager	17-29

Sample Uses for Reports	17-30
Archiving and Unarchiving Using Catalog Manager	17-30
Archiving a Folder Using Catalog Manager	17-31
Unarchiving a Folder Using Catalog Manager	17-31

Part VI Advanced Configuration Settings

18 Configuring and Managing Analyses and Dashboards

Managing Dashboards	18-1
Performing General Configuration Tasks for Analyses	18-2
Increasing Heap Size to Assist in Exports to Excel	18-2
Manually Configuring for Export	18-3
Supporting Nested Folders, Navigation, and Drill-Down	18-5
Configuring for Displaying and Processing Data in Views	18-5
Manually Configuring for Data in Views	18-6
Manually Configuring Cube Settings for Pivot Tables and Graphs	18-6
Manually Configuring Settings for Data in Views	18-7
Manually Configuring Settings for Fetching Data for Table Views, Pivot Table Views, and Trellis Views	18-11
Manually Configuring for Graphs and Gauges	18-12
Configuring Fonts for Graphs	18-14
Configuring Graph and Gauge Rendering	18-15
Manually Changing Alternating Bar Color	18-15
Manually Configuring for Interactions in Views	18-16
Configuring for Prompts	18-17
Manually Changing Presentation Settings	18-20
Manually Changing Presentation Setting Defaults	18-20
Providing Custom Links in Presentation Services	18-23
Creating the customlinks.xml File	18-23
Adding the CustomLinks Element	18-27
Setting the Custom Links Privilege	18-27
Enabling the Ability to Create Links to Dashboard Pages	18-28
Configuring an Alternate Toolbar for Oracle BI Publisher	18-29
Enabling the Ability to Export Dashboard Pages to Oracle BI Publisher	18-30
Modifying the Table of Contents for PDF Versions of Briefing Books	18-30
Configuring a Custom Download Link for the Smart View Installer	18-31
Blocking Analyses in Answers	18-32
Storing JavaScript Files	18-32
Blocking Analyses Based on Criteria	18-32
Blocking Analyses Based on Formula	18-34

Validation Helper Functions	18-35
Specifying View Defaults for Analyses and Dashboards	18-36
XML Message Files for View Defaults	18-36
Examples of Customizing Default Values for Analyses and Dashboards	18-36
Adding a Default Header or Footer to New Analyses	18-36
Preventing Auto-Previewing of Results	18-37
Setting Defaults for Analyses in the Compound Layout	18-38
Changing Dashboards Section Defaults	18-38
Specifying Dashboard Page Defaults Including Headers and Footers	18-39
Configuring for Write Back in Analyses and Dashboards	18-40
How Write Back Works	18-40
Process for Configuring Write Back	18-40
Example: Process for Configuring Write Back	18-42
Write-Back Limitations	18-43
Creating Write-Back Template Files	18-44
Setting the LightWriteback Element	18-46
Customizing the Oracle BI Web User Interface	18-47
What Are Skins and Styles?	18-47
General Tips for Customizing the Web User Interface	18-48
About Style Customizations	18-48
Modifying the User Interface Styles for Presentation Services	18-48
Approach 1: Deploying the "bicustom.ear" File for the First Time?	18-49
Approach 1: Redeploying the "bicustom.ear" File	18-51
Approach 2: Deploying Using Shared Folders	18-51
Approach 2: Viewing Your Modifications to a Shared Folder	18-54
Customizing Your Style	18-54
Example of Modifying the Skyros Master Branding Class	18-57
Embedding External Content in Dashboards	18-58

19 Configuring and Managing Agents

How Are Agents Used?	19-1
How Do Antivirus Software and Privileges Affect Agents?	19-2
Configuring Settings that Affect Agents	19-2
Manually Configuring Presentation Services Settings that Affect Agents	19-3
Manually Changing Additional Scheduler Settings that Affect Agents	19-4
What Additional Scheduler Configuration Settings Affect Agents?	19-5
General Scheduler Configuration Settings that Affect Agents	19-5
Email Scheduler Configuration Settings that Affect Agents	19-9
Agent Scheduler Configuration Settings	19-9
Controlling Delivery Options for Agents	19-11

Managing Device Types for Agents	19-11
Monitoring Active Agent Sessions	19-12

20 Configuring Advanced Options for Mapping and Spatial Information

Configuring MapViewer to Support Map Views	20-1
Manually Configuring for Map Views	20-3
Inserting Text on a Map	20-4
Configuring Maps for External Consumption	20-5

21 Configuring Resource Availability and URL Generation

Part VII Managing the Life Cycle

22 Patching Oracle Business Intelligence Systems

About Patching Oracle Business Intelligence Systems	22-1
What Is Patched for the Oracle Business Intelligence Platform?	22-2
Rolling Back a Platform Patch	22-3
Determining Current Patch Levels	22-3

23 Moving Oracle Business Intelligence Between Environments

Moving Between Environments	23-1
Moving to a New Environment	23-2
Upgrading from 11g to 12c	23-3
Migrating the Whole Server	23-3

24 Backup and Recovery of Oracle Business Intelligence Systems

Part VIII Using Oracle Essbase With Oracle Business Intelligence

25 Introduction to Using Oracle Essbase in Oracle Business Intelligence

Overview	25-1
----------	------

High-Level Roadmap for Working with Essbase When Installed with Oracle Business Intelligence	25-2
Performing Tasks When Essbase Is Installed with Oracle BI EE Compared to Performing the Same Tasks in EPM and Information on Which Guides to Consult	25-3
Installing Essbase with Oracle Business Intelligence	25-4
Installing Essbase	25-5
Selecting the Essbase Suite Option During Install	25-5
Limitations for Using Client Tools when Essbase Is Installed with Oracle Business Intelligence	25-6
Essbase Features Not Supported when Essbase Is Installed with Oracle Business Intelligence	25-6
Configuring Security for Essbase in Oracle Business Intelligence	25-7
Enabling Users to Perform Specific Actions in Essbase and Associated Tools	25-8
Enabling Drill-through Reports in Oracle BI EE 12.2.1.x	25-10
Configuring Data-Level Security Using Essbase Filters	25-11
What Are Essbase Filters?	25-12
Configuring Resource Permission Definitions for Essbase Filters	25-12
Securing Data Access with Essbase Filters	25-15
When Are Filter Access Permission Grant Changes Consumed?	25-15
How Do Filter Permission Grants Differ Between Oracle Business Intelligence and EPM?	25-16
Configuring Access to Essbase Calculations	25-16
What Are Essbase Calculations?	25-17
Configuring Resource Permission Definitions for Essbase Calculations	25-17
Enabling Users to Access Essbase Calculations	25-19
Changing Essbase Ports in Oracle Business Intelligence	25-19
Resource Permissions Reference for Essbase	25-20
What Resource Types Apply to Essbase?	25-21
What Resource Names Apply to Essbase?	25-21
What Actions Apply to Essbase?	25-22
Managing Essbase System Administration in Oracle Business Intelligence	25-24
Starting and Stopping Essbase Components	25-24
Maintaining High Availability of Essbase Components in Oracle Business Intelligence	25-24
Configuring Logging for Essbase Components	25-25
Migrating Essbase Configuration Between Domains	25-26
Monitoring Essbase Metrics	25-26
Backup and Recovery of Essbase Data	25-27
Working with Essbase Data in Oracle Business Intelligence	25-27
Enabling Single Sign-On for Essbase Data Sources	25-27
Creating, Scheduling, and Running Analyses and Reports Where Essbase Is the Data Source	25-27
Enabling Oracle BI EE to Connect to Essbase Over SSL	25-28

Part IX Reference Information

A Configuration File Settings

Configuration Files	A-1
NQSConfig.INI File Configuration Settings	A-2
About Parameters in the NQSConfig.INI File	A-3
How to Update Parameters in NQSConfig.INI	A-4
Repository Section Parameters	A-4
Multitenancy Section Parameters	A-5
MT_ROOT_DIRECTORY	A-5
MT_ENTRIES	A-5
Query Result Cache Section Parameters	A-5
ENABLE	A-6
DATA_STORAGE_PATHS	A-6
MAX_ROWS_PER_CACHE_ENTRY	A-7
MAX_CACHE_ENTRY_SIZE	A-8
MAX_CACHE_ENTRIES	A-8
POPULATE_AGGREGATE_ROLLUP_HITS	A-8
USE_ADVANCED_HIT_DETECTION	A-9
MAX_SUBEXPR_SEARCH_DEPTH	A-10
DISABLE_SUBREQUEST_CACHING	A-10
CACHE_FILE_BUFFER_SIZE	A-10
GLOBAL_CACHE_STORAGE_PATH	A-11
MAX_GLOBAL_CACHE_ENTRIES	A-11
CACHE_POLL_SECONDS	A-11
CLUSTER_AWARE_CACHE_LOGGING	A-11
General Section Parameters	A-12
LOCALE	A-12
SORT_ORDER_LOCALE	A-13
SORT_TYPE	A-14
CASE_SENSITIVE_CHARACTER_COMPARISON	A-14
NULL_VALUES_SORT_FIRST	A-16
DATE_TIME_DISPLAY_FORMAT	A-16
DATE_DISPLAY_FORMAT	A-17
TIME_DISPLAY_FORMAT	A-17
WORK_DIRECTORY_PATHS	A-17
WORK_FILE_COMPRESSION_LEVEL	A-18
ENABLE_COLUMNAR_STORAGE_FOR_WORK_FILE	A-18

WORK_DIRECTORY_SIZE_GLOBAL_LIMIT	A-18
MAX_WORK_FILE_SIZE_PERCENT	A-18
VIRTUAL_TABLE_PAGE_SIZE	A-18
USE_LONG_MONTH_NAMES	A-19
MEMORY_COMPACT_PERIOD_IN_SECONDS	A-19
USE_LONG_DAY_NAMES	A-19
USE_UPPERCASE_MONTH_NAMES	A-19
USE_UPPERCASE_DAY_NAMES	A-20
UPPERCASE_USERNAME_FOR_INITBLOCK	A-20
Security Section Parameters	A-20
DEFAULT_PRIVILEGES	A-20
PROJECT_INACCESSIBLE_COLUMN_AS_NULL	A-20
IGNORE_LDAP_PWD_EXPIRY_WARNING	A-21
MAX_AUTHENTICATION_TIME	A-21
INIT_BLOCK_LOG_TIME_THRESHOLD	A-21
NUM_INIT_BLOCK_THREADS_PER_USER	A-21
SSL	A-21
SSL_CERTIFICATE_FILE	A-22
SSL_PRIVATE_KEY_FILE	A-22
SSL_PK_PASSPHRASE_FILE	A-22
SSL_PK_PASSPHRASE_PROGRAM	A-22
SSL_VERIFY_PEER	A-22
SSL_VERIFY_SERVERS	A-22
SSL_VERIFY_CLIENTS	A-23
SSL_CA_CERTIFICATE_DIR	A-23
SSL_CA_CERTIFICATE_FILE	A-23
SSL_TRUSTED_PEER_DNS	A-23
SSL_INTERNAL_CA_CERTIFICATE_FILE	A-23
SSL_INTERNAL_TRUSTED_PEER_DNS	A-23
SSL_WEBSERVER_CA_CERTIFICATE_FILE	A-23
SSL_WEBSERVER_TRUSTED_PEER_DNS	A-24
SSL_CERT_VERIFICATION_DEPTH	A-24
SSL_CIPHER_LIST	A-24
Server Section Parameters	A-24
READ_ONLY_MODE	A-24
MAX_SESSION_LIMIT	A-25
MAX_REQUEST_PER_SESSION_LIMIT	A-25
SERVER_THREAD_RANGE	A-26
SERVER_THREAD_STACK_SIZE	A-26
DB_GATEWAY_THREAD_RANGE	A-26
DB_GATEWAY_THREAD_STACK_SIZE	A-27

HTTP_CLIENT_THREAD_RANGE	A-27
HTTP_CLIENT_THREAD_STACK_SIZE	A-27
MAX_EXPANDED_SUBQUERY_PREDICATES	A-27
MAX_QUERY_PLAN_CACHE_ENTRIES	A-28
MAX_QUERY_PLAN_CACHE_ENTRY_SIZE	A-28
MAX_DRILLDOWN_INFO_CACHE_ENTRIES	A-28
MAX_DRILLDOWN_QUERY_CACHE_ENTRIES	A-28
INIT_BLOCK_CACHE_ENTRIES	A-29
CLIENT_MGMT_THREADS_MAX	A-29
DEFAULT_JOBQUEUE_SIZE_PER_THREAD	A-29
MAX_COLUMNS_IN_SELECT	A-29
MAX_LOGICAL_DIMENSION_TABLES	A-29
MAX_LOGICAL_FACT_TABLES	A-30
MAX_LOGICAL_MEASURES	A-30
MAX_SET_OPERATION_BLOCKS	A-30
QUERY_LIMIT_WARNING_INSTEAD_OF_ERROR	A-31
RPC_SERVICE_OR_PORT	A-31
LISTEN_ADDRESS	A-31
LISTEN_PORT	A-31
ENABLE_DB_HINTS	A-32
PREVENT_DIVIDE_BY_ZERO	A-32
CLUSTER_PARTICIPANT	A-32
DISCONNECTED	A-32
AUTOMATIC_RESTART	A-33
VARIABLE_VALUE_LIMIT	A-33
EVALUATE_SUPPORT_LEVEL	A-33
FMW_SECURITY_SERVICE_URL	A-33
FMW_SECURITY_SERVICE_MAX_NUMBER_OF_CONNECTIONS	A-34
FMW_SECURITY_SERVICE_MAX_NUMBER_OF_RETRIES	A-34
ENABLE_NUMERIC_DATA_TYPE	A-34
ENDECA_SERVLET_URL	A-34
High Availability Parameters	A-35
HA_DB_PING_PERIOD_MILLISECS	A-35
Dynamic Library Section Parameters	A-35
Usage Tracking Section Parameters	A-36
ENABLE	A-36
DIRECT_INSERT	A-37
STORAGE_DIRECTORY	A-38
CHECKPOINT_INTERVAL_MINUTES	A-38
FILE_ROLLOVER_INTERVAL_MINUTES	A-39
CODE_PAGE	A-39

PHYSICAL_TABLE_NAME	A-39
CONNECTION_POOL	A-40
INIT_BLOCK_TABLE_NAME	A-40
INIT_BLOCK_CONNECTION_POOL	A-41
BUFFER_SIZE	A-41
BUFFER_TIME_LIMIT_SECONDS	A-41
NUM_INSERT_THREADS	A-41
MAX_INSERTS_PER_TRANSACTION	A-41
JOBQUEUE_SIZE_PER_INSERT_THREADPOOL_THREAD	A-42
THROW_INSERT_WHEN_JOBQUEUE_FULL	A-42
SUMMARY_STATISTICS_LOGGING	A-42
SUMMARY_ADVISOR_TABLE_NAME	A-43
Query Optimization Flags Section Parameters	A-43
STRONG_DATETIME_TYPE_CHECKING	A-43
MDX Member Name Cache Section Parameters	A-43
ENABLE	A-44
DATA_STORAGE_PATH	A-44
MAX_SIZE_PER_USER	A-44
MAX_MEMBER_PER_LEVEL	A-44
MAX_CACHE_SIZE	A-44
Aggregate Persistence Section Parameters	A-44
AGGREGATE_PREFIX	A-44
AGGREGATE_THREAD_POOL_SIZE	A-45
AGGREGATE_AW_NAME	A-45
PREAGGREGATE_AW_CUBE	A-45
SUPPORT_ANALYTICAL_WORKSPACE_TARGETS	A-45
JavaHost Section Parameters	A-45
JAVAHOST_HOSTNAME_OR_IP_ADDRESSES	A-45
JAVAHOST_HOSTNAME_OR_IP_ADDRESSES_OVERRIDE	A-46
Datamart Automation Section Parameters	A-46
ESSBASE_SERVER	A-46
DMA_DATABASE	A-46

B Advanced Configuration Reference

Making Advanced Configuration Changes for Presentation Services	B-1
Protecting Pages in Oracle BI EE from Attack	B-5
Using the JavaHost Service for Oracle BI Presentation Services	B-6

C Mapping User Interface Labels with Configuration File Elements

D BI-Specific WLST Command Reference

Preface

The Oracle Business Intelligence Foundation Suite is a complete, open, and integrated solution for all enterprise business intelligence needs, including reporting, ad hoc queries, OLAP, dashboards, scorecards, and what-if analysis. The Oracle Business Intelligence Foundation Suite includes Oracle BI Suite.

Oracle BI Suite (Oracle BI EE) is a comprehensive set of enterprise business intelligence tools and infrastructure, including a scalable and efficient query and analysis server, an ad-hoc query and analysis tool, interactive dashboards, proactive intelligence and alerts, and an enterprise reporting engine.

The components of Oracle BI EE share a common service-oriented architecture, data access services, analytic and calculation infrastructure, metadata management services, semantic business model, security model and user preferences, and administration tools. Oracle BI EE provides scalability and performance with data-source specific optimized request generation, optimized data access, advanced calculation, intelligent caching services, and clustering.

This guide contains information about system administration tasks and includes topics on starting and stopping processes, managing logging and usage tracking, managing query caching and performance, managing scalability and high availability, and setting configuration options.

Audience

This document is intended for system administrators who are responsible for managing Oracle Business Intelligence processes, logging, caching, monitoring, high availability, and configuration.

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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documentation and Other Resources

Learn more about the system and related tasks, software, and information.

See the Oracle Business Intelligence documentation library for a list of related Oracle Business Intelligence documents.

In addition:

- Go to the Oracle Learning Library for Oracle Business Intelligence-related online training resources.
- Go to the Product Information Center support note (Article ID 1267009.1) on My Oracle Support at <https://support.oracle.com>.

Conventions

The following text conventions are used in this document.

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	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.

New Features for Oracle Business Intelligence System Administrators

This section describes changes to system administration features for Oracle Business Intelligence Enterprise Edition 12c (12.2.1).

This section contains the following topics:

- [New Features and Changes for Oracle BI EE 12 c \(12.2.1.3.0\)](#)
- [New Features and Changes for Oracle BI EE 12 c \(12.2.1.2.0\)](#)
- [New Features and Changes for Oracle BI EE 12c \(12.2.1.1.0\)](#)
- [New Features and Changes for Oracle BI EE 12c \(12.2.1.0\)](#)

New Features and Changes for Oracle BI EE 12 c (12.2.1.3.0)

New system administration features and changes are included in Oracle BI EE 12c (12.2.1.3).

There are no new features this release.

New Features and Changes for Oracle BI EE 12 c (12.2.1.2.0)

New system administration features and changes are included in Oracle BI EE 12c (12.2.1.2).

There are no new features in this release.

New Features and Changes for Oracle BI EE 12c (12.2.1.1.0)

New system administration features and changes are included in Oracle BI EE 12c (12.2.1.1).

There are no new features in this release.

New Features and Changes for Oracle BI EE 12c (12.2.1.0)

New system administration features and changes are included in Oracle BI EE 12c (12.2.1.0)

If you are upgrading to Oracle BI EE 12c (12.2.1.0) from Oracle BI EE 11g (11.1.1.9), then read the following information carefully, because there are significant differences in features, tools, and procedures. For more information about upgrading to Oracle BI EE 12c, see *Upgrading Oracle Business Intelligence*.

These features and changes include:

- [Invoking WLST From a Single Location](#)
- [Oracle Home Location Redefined and No Middleware Home](#)
- [OPMN is No Longer Used in Fusion Middleware](#)
- [Oracle Web Cache Not Part of Fusion Middleware](#)
- [Moving From Test To Production is Carried Out in a Different Way](#)
- [New Commands For Process Control](#)
- [Managing Metadata In Business Intelligence Archive Files](#)
- [Single Enterprise Install](#)
- [Changes to Scaling Out](#)
- [Simplified Configuration](#)
- [Managing System Component Instances Using Commands](#)
- [Collecting Diagnostic Bundles](#)

Invoking WLST From a Single Location

In previous releases, you invoked WLST from different locations, depending on whether you were using the commands for Oracle WebLogic Server, system components, or Java components such as Oracle SOA Suite. In this release, you invoke WLST from:

```
(UNIX) ORACLE_HOME/oracle_common/common/bin/wlst.sh  
(Windows) ORACLE_HOME\oracle_common\common\bin\wlst.cmd
```

See [Using the WebLogic Scripting Tool \(WLST\)](#).

Oracle Home Location Redefined and No Middleware Home

Redefining of the Oracle home and elimination of the Middleware home. See “New and Deprecated Terminology for 12c” in *Understanding Oracle Fusion Middleware*.

OPMN is No Longer Used in Fusion Middleware

OPMN is no longer used in Oracle Fusion Middleware. Instead, system components are managed by the WebLogic Management Framework, which includes WLST, Node Manager and pack and unpack. See [What is the WebLogic Management Framework](#) in *Understanding Oracle Fusion Middleware*.

Oracle Web Cache Not Part of Fusion Middleware

Oracle Web Cache is no longer part of Oracle Fusion Middleware.

Moving From Test To Production is Carried Out in a Different Way

The test to production operation is still possible however, the process is different from what was available in Oracle Business Intelligence Release 1 (11.1.1) as it applies

solely to metadata (content, data model and authorization). See [Moving Oracle Business Intelligence Between Environments](#).

New Commands For Process Control

New process control commands replace the old start stop commands. See [Process Control Commands](#).

Managing Metadata In Business Intelligence Archive Files

All Oracle Business Intelligence metadata, including repository, Presentation Services catalog, and user authentication is stored in BAR archive files. The BAR file is a mechanism for managing or moving a self contained set of Oracle BI metadata between environments. See [Managing Metadata and Working with Service Instances](#) .

Single Enterprise Install

In this release the Oracle Universal installer offers a single install type for your Enterprise which provides an Administration server, and a Managed server. See [What Is the Oracle Business Intelligence System Logical Architecture?](#) and [Installing and Configuring Oracle Business Intelligence](#).

Changes to Scaling Out

In this release the scale out procedures for Oracle Business Intelligence have changed. See [Scaling Your Deployment](#) .

Simplified Configuration

Configuration files are no longer duplicated. Separate configuration files still exist for example, for Oracle BI Presentation Services and BI Server, but they are not duplicated in the case of a cluster. See [Configuring Oracle Business Intelligence System Settings](#).

Managing System Component Instances Using Commands

OBIS (BI Server) system component instances are separately managed in BI 12.2.1 using service instance commands. See [Managing Service Instances](#).

Collecting Diagnostic Bundles

A new script enables you to collect the diagnostic bundles needed by Oracle Support or Development to help resolve issues. See [Collecting Diagnostic Bundles](#).

Synchronizing Mid-Tier Database Connection Details Command

A new command enables you to synchronize mid-tier database connection details when they have changed. See [BI-Specific WLST Command Reference](#).

Part I

Administering Oracle Business Intelligence

This part introduces administering the Oracle Business Intelligence system.

- [Introduction to Oracle Business Intelligence System Administration](#)

1

Introduction to Oracle Business Intelligence System Administration

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

This chapter includes the following sections:

- [What Are the Oracle Business Intelligence System Administration Tasks?](#)
- [Getting Started with Managing Oracle Business Intelligence](#)
- [What Is the Oracle Business Intelligence System Logical Architecture?](#)
- [Key Directories in Oracle Business Intelligence](#)
- [What System Administration Tools Manage Oracle Business Intelligence?](#)
- [Working with the Sample Application](#)
- [Oracle BI Publisher Integration](#)
- [Topics of Interest in Other Guides](#)
- [System Requirements and Certification](#)

What Are the Oracle Business Intelligence System Administration Tasks?

System administrators need to take several steps to configure Oracle Business Intelligence properly.

Administering an Oracle Business Intelligence 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 the Oracle Business Intelligence 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 Oracle Business Intelligence 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 [Getting Started with Managing Oracle Business Intelligence](#).

Getting Started with Managing Oracle Business Intelligence

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 in Oracle Business Intelligence and indicates where to find related information.

System Administration Task	More Information
Learning about Oracle Business Intelligence 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 Oracle Business Intelligence status	Displaying Oracle Business Intelligence Pages in Fusion Middleware Control Also contains information about using Fusion Middleware Control and using WebLogic Server Administration Console.
Configuring Oracle Business Intelligence	Configuring Oracle Business Intelligence System Settings Contains information about the available methods for updating configuration settings and where configuration files are located.
Starting and stopping Oracle Business Intelligence	Managing Oracle Business Intelligence Processes Contains various topics on starting and stopping components, in addition to troubleshooting information.
Managing availability and capacity	Scaling and Deploying 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	Resolving Issues Contains chapters about diagnosing and resolving issues and about usage tracking.

System Administration Task	More Information
Configuring Oracle Business Intelligence	Configuring Oracle Business Intelligence Contains chapters about required configuration such as configuring repositories 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	<i>Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition</i>
Managing the life cycle.	Managing the Life Cycle Contains chapters about life cycle management tasks such as patching, moving between environments, and backup and recovery.
Using Essbase with Oracle Business Intelligence	Using Oracle Essbase With Oracle Business Intelligence Contains a chapter about using Essbase when it is installed with Oracle Business Intelligence.
Securing the system	<ul style="list-style-type: none"> • Defines administrative role membership • Secures middle-tier communications Secure Sockets Layer (SSL) and Single Sign-On (SSO) are not described in this guide. See <i>Security Guide for Oracle Business Intelligence Enterprise Edition</i> .

What Is the Oracle Business Intelligence System Logical Architecture?

The Oracle Business Intelligence system logical architecture comprises a single integrated set of manageable components called the Oracle 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 Oracle Business Intelligence, such as Oracle HTTP Server or Apache HTTP Server. A web server is not included by default in the Oracle Business Intelligence installer and is not part of the Oracle Business Intelligence system logical architecture. You must install and configure a web server separately.

This section contains the following topics:

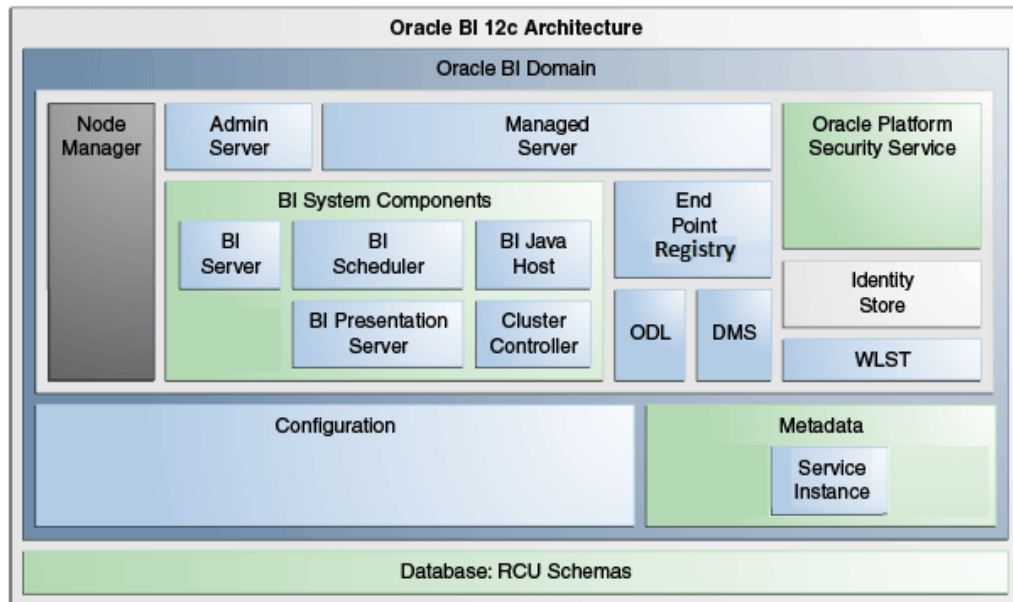
- [Oracle Business Intelligence System Architecture.](#)
- [Oracle Business Intelligence Components.](#)
- [About the Administration Server, Managed Servers, and System Components.](#)

Oracle Business Intelligence System Architecture

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

See [Scaling Your Deployment](#) .

The figure below illustrates the Oracle Business Intelligence system architecture on a single host. See [Oracle Business Intelligence Components](#).



Oracle Business Intelligence is installed on a single host but can be scaled 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 will be active on only one host.

Oracle Business Intelligence Components

When you install Oracle Business Intelligence, you can install several components in the Oracle 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, metadata repositories, and infrastructure.

- **Admin 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 Oracle BI Publisher, Visual Analyzer, Presentation Services, and Composer.

An Oracle 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 and provide the core services that enable Oracle Business Intelligence.

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 an Oracle Business Intelligence system.

For more information about:

- Security configuration - See *Security Guide for Oracle Business Intelligence Enterprise Edition*.
- Metadata - See [Managing Metadata and Working with Service Instances](#) and *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

About the Administration Server, Managed Servers, and System Components

Oracle Business Intelligence 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*.

For information about Essbase when installed with Oracle Business Intelligence, see [What System Administration Tools Manage Oracle Business Intelligence?](#).

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 Oracle Business Intelligence 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 System Administration Tools Manage Oracle Business Intelligence?](#).

- **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.
 - **Visual Analyzer** — This component provides a simple-to-use visual analytical interface.
 - **BI 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 Oracle BI Presentation Catalog, to invoke analyses, agents, and conditions. These services make it easy to invoke Oracle Business Intelligence functionality from Business Process Execution Language (BPEL) processes.
 - **Presentation Services** — This component is a JEE application that routes HTTP and SOAP requests to Oracle BI 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 at the heart of Oracle Business Intelligence and provides 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. (Oracle BI 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, Oracle BI Publisher, and graph generation. It also enables Oracle 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 an Oracle BI Presentation Catalog service on the file system 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 in Oracle Business Intelligence

There are three key top-level directories in Oracle Business Intelligence 12c.

- **ORACLE_HOME** - for binaries.
There is one ORACLE_HOME for each host, or mounted from shared storage.
- **DOMAIN_HOME** - for configuration, and logs.
There is one DOMAIN_HOME for each host (also referred to as BI_DOMAIN, or bidomain).
- **SDD (Singleton Data Directory)** - for metadata and other 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 have just created a domain on one host, then SDD is set to *DOMAIN_HOME/bidata*.

If you have scaled-out, the SDD changes to use mounted shared storage. In this case, the SDD will not be *DOMAIN_HOME/bidata*.

See [Changing the Singleton Data Directory \(SDD\)](#).

What System Administration Tools Manage Oracle Business Intelligence?

There are several tools that you can use to manage Oracle Business Intelligence.

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

Tool	Purpose	More Information
Fusion Middleware Control	Monitor, manage, and configure system components for Oracle Business Intelligence.	Fusion Middleware Control
Oracle WebLogic ServerAdministration Console	Monitor and manage JEE Java components for Oracle Business Intelligence.	Oracle WebLogic Server Administration Console
Process control command line tool	Manage system components for Oracle Business Intelligence (for advanced users).	Process Control Commands
Oracle WebLogic Scripting Tool (WLST)	Programmatically administer Oracle Business Intelligence.	Oracle WebLogic Scripting Tool (WLST)

Tool	Purpose	More Information
Oracle BI Administration Tool	Manage the metadata repository for Oracle Business Intelligence.	Oracle BI Administration Tool
Catalog Manager	Manage the Oracle BI Presentation Catalog online and offline.	Catalog Manager
Job Manager	Manage the Oracle BI Scheduler	Job Manager

Fusion Middleware Control

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

Fusion Middleware Control is available only if the Administration Server is running, as described in [Conditions for Starting the Oracle Business Intelligence System](#).

See [Using Tools for Managing and Configuring Oracle Business Intelligence](#).

Oracle WebLogic Server Administration Console

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

See [Using Tools for Managing and Configuring Oracle Business Intelligence](#).

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 Business Intelligence 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 Business Intelligence system components:

- A command-line interface for advanced users to control Oracle Fusion Middleware components.
For information, see [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#).
- An integrated way to manage Oracle Business Intelligence component processes.

Oracle WebLogic Scripting Tool (WLST)

The Oracle WebLogic Scripting Tool (WLST) is a command-line scripting environment (for advanced administrator use), which enables you to programmatically administer Oracle Business Intelligence.

The WLST scripting environment is based on the Java scripting interpreter Jython. You can use this tool interactively on the command line, in batch scripts that are supplied in a file (Script Mode, where scripts invoke a sequence of WLST commands without requiring your input), or embedded in Java code. You can extend the WebLogic scripting language by following the Jython language syntax. See [Using the WebLogic Scripting Tool \(WLST\)](#) and *WLST Command Reference for WebLogic Server*.

Oracle BI Administration Tool

The Oracle BI Administration Tool enables you to manage the metadata repository.

See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Catalog Manager

The Catalog Manager is a Windows tool that is the interface with the Oracle BI Presentation Catalog.

Through Catalog Manager, you can perform online and offline management of Oracle BI Presentation Catalogs. See [Configuring and Managing the Oracle BI Presentation Catalog](#).

Job Manager

The Job Manager is a Windows tool that is the interface with the Oracle BI Scheduler.

Through Job Manager, you can connect to, start and stop the Oracle BI Scheduler, add and manage jobs, and manage job instances. For information, see *Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*.

Working with the Sample Application

The sample application allows you to test configurations and procedures before promoting them to a live scenario.

You can configure Oracle Business Intelligence with a simplified sample application. This application is often referred to as SampleApp Lite.

You can also download and configure a more robust sample application. Instructions for this configuration are available at the following location:

<http://www.oracle.com/technetwork/middleware/bi-foundation/obiee-samples-167534.html>

See About the SampleApp Demonstration Repository in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for information about the sample repository that is provided with Oracle Business Intelligence.

Oracle BI Publisher Integration

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

For information about running BI Publisher, see *User's Guide for Oracle Business Intelligence Publisher*.

Topics of Interest in Other Guides

Some topics that might be of interest to system administrators are covered in other guides.

The following table lists these topics and indicates where to go for more information.

Topic	Where to Go for More Information
Third-party tools and relational data source adapters.	System Requirements and Certification
Configuration tasks for Oracle BI Scheduler.	<i>Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition</i>
Configuring data sources.	<i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>
Security, including configuring SSO and SSL.	<i>Security Guide for Oracle Business Intelligence Enterprise Edition</i>

Topic	Where to Go for More Information
Installing and upgrading.	<i>Installing and Configuring Oracle Business Intelligence Upgrading Oracle Business Intelligence</i>
Configuring comments and status overrides in Oracle Scorecard and Strategy Management.	<i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>
Converting Oracle Business Intelligence proprietary metadata to an XML file and importing the metadata into your Oracle database.	<i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>
Propagating UI hints (labels and tooltips) from an ADF data source to display in Oracle BI Answers.	<i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>

System Requirements and Certification

System requirements and certification are described in their own documents.

Refer to the system requirements and certification documentation for information about hardware and software requirements, platforms, databases, and other information. Both of these documents are available on Oracle Technology Network (OTN).

The system requirements document covers information such as hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches:

<http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-requirements-100147.html>

The certification document covers supported installation types, platforms, operating systems, databases, JDKs, and third-party products:

<http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html>

Part II

Managing Processes and Components

This part explains how to manage processes and components in Oracle Business Intelligence. It includes the following chapter:

- [Managing Oracle Business Intelligence Processes](#)

2

Managing Oracle Business Intelligence Processes

This chapter describes managing Oracle Business Intelligence processes and includes the following sections:

- [About Managing Oracle Business Intelligence Processes](#)
- [Conditions for Starting the Oracle Business Intelligence System](#)
- [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#)
- [Using Fusion Middleware Control to Start and Stop BI System Component Processes](#)
- [Using Fusion Middleware Control to Start and Stop Java Components](#)
- [Using Oracle WebLogic Server Administration Console to Start and Stop Java Components](#)

About Managing Oracle Business Intelligence Processes

System administrators start and stop the Oracle Business Intelligence 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 Oracle Business Intelligence system is an essential activity when diagnosing and resolving availability and performance issues, and when performing life-cycle and management operations. See [Diagnosing and Resolving Issues in Oracle Business Intelligence](#).

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

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

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

For information about the Oracle Business Intelligence installed components, see [Oracle Business Intelligence Components](#).

Conditions for Starting the Oracle Business Intelligence System

Starting the Oracle Business Intelligence system begins with the Administration Server, then the Managed Servers, and the system components.

If the computer that hosts the Administration Server is not running or has been rebooted, then you must ensure that the computer is running and you must start the Oracle Business Intelligence system.

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

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

Using Commands to Start, Stop, and View Status of Oracle BI EE Processes

Several components of the software can be controlled using script commands.

You use script commands to start, stop, and view status of Oracle Business Intelligence components.

- [Starting Oracle Business Intelligence Component Processes in a Domain](#)
- [Stopping Oracle Business Intelligence Component Processes in a Domain](#)
- [Viewing the Status of Oracle Business Intelligence Components in a Domain](#)

Starting Oracle Business Intelligence Component Processes in a Domain

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

Assumptions

- The start command starts Node Manager locally and remotely (on clustered servers) if not already running.
- The start command runs only from the master host.
- The start command does not 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 do not require credentials.

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

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

```
DOMAIN_HOME/bitools/bin
```

On UNIX | 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`

Argument	Description
<code>-h<Domain home></code>	(optional) Enables you to specify the domain home, includes the <i>domainName</i> directory. Default is <i>DOMAIN_HOME</i> if set.
<code>-i <startServersList></code>	(optional) Enables you 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.
<code>-r<restartIntervalSeconds></code>	(optional) Specifies the number of seconds during which the system components can be restarted. Default is 3600, Maximum is 214748647, Minimum is 300.
<code>-m<restartMaxValue></code>	(optional) Specifies the number of times that the Node Manager can restart the System Components within the interval specified in Restart Interval in Seconds. Default is 2, Maximum is 2147483647, Minimum is 0. If set to 0, then auto restart of system components is disabled.
<code>-c<Clear cached credentials></code>	(optional) Clears and resets the cached WebLogic administrator credentials, prompts for user name and password. Use this parameter after changing the default WebLogic Server administrator password. The password is encrypted and cached as before, for later use in start/stop, without user interaction. Default is false. The Node Manager credentials are not changed.

If no instances are specified as arguments in the command, the administration server, managed server, and all system components, are started by default.

2. A list of the inactive components to be started is displayed.
3. Components are started.

If you don't specify `-i`, then start will start all inactive processes. It does not 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.

Stopping Oracle Business Intelligence Component Processes in a Domain

This section describes 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 master host.
- The stop command continues until all specified component processes are shutdown.
- The stop command initially prompts for credentials and automatically creates a boot identity file, so that subsequent runs do not require credentials.

- Stopping specific process may cause failover, so familiarize yourself with [Scaling 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 WebLogic Server.

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

`DOMAIN_HOME/bitools/bin`

On UNIX | Windows:

```
./stop.sh | stop.cmd {-i <list of instances>} {-c}
```

For example, `./stop.sh -i obisl,obips1`

Argument	Description
<code>-h<Domain home></code>	(optional) Enables you to specify the domain home includes the <code>domainName</code> directory. Default is <code>DOMAIN_HOME</code> if set.
<code>-i <list of instances></code>	(optional) Enables you 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.
<code>-c<Clear cached credentials></code>	(optional) Clears and resets the cached WebLogic administrator credentials, prompts for username and password. Use this parameter after changing the default WebLogic Server administrator password. The password is encrypted and cached as before, for later use in start/stop, without user interaction. Default is false. Node Manager credentials are not changed.

If no instances are specified as arguments in the command, the administration server, managed server and all system components are shutdown by default.

2. Components are shut down.

Viewing the Status of Oracle Business Intelligence Components in a Domain

The status command displays a status report for components within a domain.

Assumptions

- The status command reports node manager status.
- The status command only runs from the Master 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 do not 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 UNIX | Windows:

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

where {-v} is verbose

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

Using Fusion Middleware Control to Start and Stop BI System Component Processes

If the Oracle Business Intelligence system has been started, then you can start, stop, and restart the Oracle Business Intelligence system component processes, using Fusion Middleware Control.

If Fusion Middleware Control is not available, then see [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#).

1. Go to the Overview page, as described in [Displaying Oracle Business Intelligence 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.

The screenshot shows the Oracle Enterprise Manager Fusion Middleware Control 12c interface. The top navigation bar includes the Oracle logo, 'Enterprise Manager Fusion Middleware Control 12c', and various utility icons like 'WebLogic Domain', 'Setup', and 'weblogic'. The main content area is for a 'Business Intelligence Instance' (biinstance). The 'Availability' tab is active, and the 'Processes' sub-tab is selected. Below the sub-tabs, there are buttons for 'Start All', 'Stop All', 'Restart All', 'Start Selected', 'Stop Selected', and 'Restart Selected'. A table lists the following processes, all with a green up arrow in the 'Status' column:

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 Oracle Business Intelligence processes:

- [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#)

- [Using Oracle WebLogic Server Administration Console to Start and Stop Java Components](#)

Using Fusion Middleware Control to Start and Stop Java Components

Use this topic to monitor status and start and stop Oracle Business Intelligence 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 [Logging 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.

The screenshot shows the WebLogic Domain home page. On the left, there are three summary cards: 'Servers' (2 Down, 1 Up), 'Clusters' (1 Down), and 'Deployments' (61 Down, 6 In). The main content area is titled 'Administration Server' and shows details for 'AdminServer' (Name: AdminServer, Host: example.com, Listen Port: 7001). Below this is a 'Servers' table with the following data:

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

At the bottom of the table, it shows 'Columns Hidden: 4' and '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 Oracle Business Intelligence 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 Oracle Business Intelligence Java components. See [Managing Oracle Business Intelligence Java Components Using the Oracle WebLogic Server Administration Console](#).

3. Using the WebLogic Domain home page, you can perform the following Oracle Business Intelligence 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 [Using 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 [Managing Oracle Business Intelligence Java Components Using the Oracle WebLogic Server Administration Console](#).

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

In the event the standard methods for starting and stopping Java components cannot be used, you can use the Oracle WebLogic Server Administration Console.

It is not recommended to use Oracle WebLogic Server Administration Console to start and stop Java components. You can also use or Fusion Middleware Control to start and stop Java components (see [Using Fusion Middleware Control to Start and Stop BI 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 required, as shown in the following illustration.

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: bi

Home > bi > Summary of Deployments

Summary of Deployments

Configuration **Control** Monitoring

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.

Customize this table

Deployments

Start Stop Showing 1 to 10 of 29 Previous Next

<input type="checkbox"/>	Name	State	Health	Type	Targets	Scope	Domain Partitions
<input checked="" type="checkbox"/>	adminservice (11.1.1)	Active	OK	Enterprise Application	bi_cluster	Global	
<input type="checkbox"/>	analytics	Active	OK	Enterprise Application	bi_cluster	Global	

Part III

Scaling and Deploying for High Availability and Performance

Learn how to manage deployment, availability, and capacity for Oracle Business Intelligence.

See the following topics for more information:

- [Scaling Your Deployment](#)
- [Deploying Oracle Business Intelligence for High Availability](#)
- [Managing Performance Tuning and Query Caching](#)

3

Scaling Your Deployment

This chapter describes how to manage the capacity and availability of your deployment of Oracle Business Intelligence. By default, Oracle Business Intelligence 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.

This chapter includes the following sections:

- [About Scaling Oracle Business Intelligence](#)
- [Setting Up Shared Files and Directories](#)
- [Managing Capacity in Oracle Business Intelligence \(Vertically Scaling\)](#)
- [Managing Availability in Oracle Business Intelligence \(Horizontally Scaling\)](#)
- [Validating That Your System Has Been Scaled Correctly](#)

About Scaling Oracle Business Intelligence

Scaling is the process of increasing or decreasing the capacity of the system by changing the number of processes available to service requests from Oracle Business Intelligence 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 Oracle Business Intelligence environment applies principally to resource-intensive system processes and Java components. When you deploy more processes, Oracle Business Intelligence can handle more requests while staying responsive to requests.

Vertical scaling involves adding more Oracle Business Intelligence components to the same computer, to make increased use of the hardware resources on that computer. For example, Oracle Business Intelligence can be vertically scaled by increasing the number of system components servicing requests on a given computer and results in increased use of the hardware resources on a given computer.

Horizontal scaling involves adding more computers to the environment. For example, Oracle Business Intelligence is horizontally scaled by distributing the processing of requests across multiple computers.

You can scale both Oracle Business Intelligence Java components and system components. See [About the Administration Server, Managed Servers, and System Components](#).

The three system components that support both horizontal and vertical scale-out are Oracle BI 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 does not 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 do not 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 [Monitoring Service Levels](#) for more information about viewing system metrics.

You also must scale out processes to achieve redundancy when you want to configure a highly available Oracle Business Intelligence environment. See [Deploying Oracle Business Intelligence for High Availability](#).

What Processes Should I Scale?

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

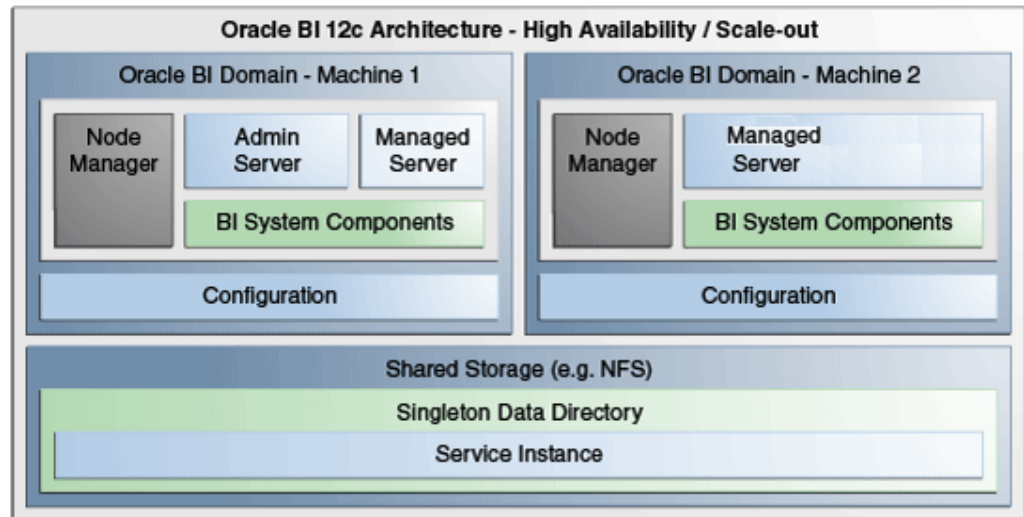
Follow these guidelines for scaling Managed Servers and system components:

- Ensure that you run at least one Managed Server on each computer in the deployment. During installation the Oracle Business Intelligence Configuration Assistant provisions one Managed Server. Do not disable or remove it.
- Do not 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 do not have a significant performance impact.
- You can decide based on observed load which system components to run on each computer. You can have 0 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 do not 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.

Setting Up Shared Files and Directories

When you have multiple instances of a given Oracle Business Intelligence 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 (see the following illustration) including scale out of Oracle Business Intelligence components.



This section contains the following topics:

- [Changing the Singleton Data Directory \(SDD\)](#)
- [Setting Up the Global Cache](#)

Changing the Singleton Data Directory (SDD)

Oracle Business Intelligence 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 in Oracle Business Intelligence](#).

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

For example, on Windows:

```
dir \\example.com\dir
```

For example, on UNIX:

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

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

```
DOMAIN_HOME/bitools/bin
```

For example on UNIX enter:

```
./stop.sh
```

3. Backup 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.
7. Start all Oracle Business Intelligence processes by running the following command located in:

```
DOMAIN_HOME/bitools/bin/
```

For example on UNIX enter:

```
./start.sh
```

The SDD is now configured for all host computers.

Setting 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 is 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 Business Intelligence Instance Configuration page in Fusion Middleware Control to set the **Global cache path** and **Global cache size** options.

See [Using Fusion Middleware Control to Set Global Cache Parameters](#).

Managing Capacity in Oracle Business Intelligence (Vertically Scaling)

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

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

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

The commands described in this section should only be used by advanced users.

- [Adding System Components](#)
- [Removing System Components](#)

Adding System Components

You can add BI System Components to a computer when the system is stopped (offline).

Note:

If SSL is configured, see [Configuring SSL in Oracle Business Intelligence in Security Guide for Oracle Business Intelligence Enterprise Edition](#)

Assumptions:

- You must have appropriate file system permissions.
- Ports are allocated from the Oracle Business Intelligence 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 is not 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 it is required to add another instance on another host, then an existing instance must first be removed.

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 UNIX:

```
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 UNIX:

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

For example on Windows:

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

Command	Description
createOBICCComponent(domainHome, machine, port=None, portMonitor=None)	This command creates a new cluster component.
createOBISCHComponent(domainHome, machine, port=None, portMonitor=None, portScript=None)	This command creates a new scheduler component.
createOBIPSComponent(domainHome, machine, port=None)	This command creates a new BI Presentation Server component.
createOBIJHComponent(domainHome, machine, port=None)	This command creates a new JavaHost component
listBISystemComponents(domainHome)	This command returns an object listing all of the system components in the domain.
getBISystemComponents(domainHome, instanceId)	This command displays details of system component with specified instanceId.

- Once command(s) have completed successfully, then you must commit the changes using the `updateDomain('DOMAIN_HOME')` command.

The WLST command is same for all environments.

For example:

```
updateDomain()
```

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

The WLST command is same for all environments.

For example:

```
closeDomain()
```

- Start the new component in:

`DOMAIN_HOME/bitools/bin/`

For example, enter:

```
./start.sh
```

See [Starting Oracle Business Intelligence Component Processes in a Domain](#).

 **Note:**

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

Post Conditions

- The new component is created.
- New port(s) are allocated.

- The new component is started.
- See [Using the WebLogic Scripting Tool \(WLST\)](#).

Removing System Components

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

Assumptions:

- Run commands when 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 [About System Components](#).

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

```
DOMAIN_HOME/bitools/bin/
```

For example on UNIX enter:

```
./stop.sh
```

See [Stopping Oracle Business Intelligence 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 system component(s) and un-allocates ports.

See [Using the WebLogic Scripting Tool \(WLST\)](#).

The deleted system component name is displayed.

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

```
DOMAIN_HOME/bitools/bin/
```

For example on UNIX enter:

```
./start.sh
```

See [Starting Oracle Business Intelligence Component Processes in a Domain](#).

Managing Availability in Oracle Business Intelligence (Horizontally Scaling)

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

See [Setting 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*. If a front end load balancer has been setup, then you must set the WebLogic Server Front-End Host and Port for the BI cluster (bi_cluster).

The commands described in this section should only be used by advanced users. Note that availability commands automatically create a symmetric set of processes when configuring additional hosts.

See [Using the WebLogic Scripting Tool \(WLST\)](#).

**Note:**

To add a new node to a cluster when SSL has been configured, you must scale out to the new cluster (see [Adding New Computers](#)), then ensure SSL is set up correctly (see *Configuring SSL in Oracle Business Intelligence in Security Guide for Oracle Business Intelligence Enterprise Edition*).

- [Adding New Computers](#)
- [Removing Existing Computers](#)

Adding New Computers

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

Assumptions

- The new computer must meet the same install pre-requisites (for example, operating system, memory).
- SDD must have been set up.
- `ORACLE_HOME` must be the same absolute path on both computers.
- It is recommended, but not required, that the `DOMAIN_HOME` is the same on both hosts.
- A symmetric set of active-active components is created on the new computer.
- The BI system must be stopped (offline).
- You must have appropriate file system (offline) or Weblogic Administrator (online) permissions.
- The same ports are allocated as on the original host.
- The managed server is added to the existing Oracle Business Intelligence cluster.
- Cluster Controller, Scheduler and BI Server mastership is not changed.
- Note that optional base computer and server parameters are provided to support the case where m1/bi_server1 has been deleted.
- Unless provided the computer (WebLogic machine) name will default to the listen address and must be less than 32 characters.

Adding a new computer creates an additional managed server, node manager, system components, and services on the new computer.

1. Shutdown all services on the master computer (node 1) before running the clone script:

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

2. On node 1, run the clone script:

The script creates the Jar file containing the domain directory and updates the config.xml file in node 1 with the new host name, Managed Server and BI System Components.

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

<new machine name> is optional and defaults to the listen address.

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 (node 2), install WebLogic Server and Oracle Business Intelligence.
See *Installing and Configuring Oracle Business Intelligence*.
4. Test connectivity between the two hosts.
5. Copy the Jar file from the master computer (node 1 with the Admin server) to the new computer (node 2).
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 master node] -domain=DOMAIN_HOME -
nodemanager_type=PerDomainNodeManager
```

Note:

The syntax to display help is: `./unpack.sh|cmd -help`.

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:

On node 1: Node Manager, Admin server, Managed server for node 1 and BI System Components for node 1.

On node 2: Node Manager and Managed server (this is so the config.xml can be created).

The BI System Components for node 2 should remain down for now.

 **Note:**

If Node Manager on node 2 is started, then all other components can be started using the same script from node 1. So you can start the components in this order:

- On node 2:
Node Manager for node 2
- On node 1:
Node Manager for node 1.
Admin server.
Managed server for node 1
BI system components for node 1

For example:

Node 2:

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

Node 1:

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

8. Re-synchronize the data source on new machine (node 2).

On node 2 run the following script:

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

See [BI-Specific WLST Command Reference](#).

9. On the master host computer (node 1), start the new BI System Components for node 2:

```
DOMAIN_HOME/bitools/bin/
```

Enter:

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

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

See [Starting Oracle Business Intelligence Component Processes in a Domain](#).

Post Conditions

- Computer, Managed server, Node Manager and System Components are created.
- Service instances are registered on the second computer.
- Ports are allocated.

 **Note:**

If you're upgrading to version 12.2.1.2 or later, after horizontally scaling your system you may encounter an error when uploading your repository (using the `uploadrpd` command) similar to:

```
Operation failed.
An exception occurred during execution, please check server logs.
uploadRpd Failed: Did not receive a success response to REST call.
```

The log will display an error similar to:

```
Aug 16, 2017 22:10:32 oracle.bi.lcm.rest.si.rpd.RpdEndpointV1
SEVERE - Exception during RPD file upload: java.io.IOException: No such
file or directory
```

If this occurs, create a folder on the master machine: `ORACLE_HOME/user_projects/domains/DOMAIN_HOME/bidata/service_instances/<serviceinstanceKey>/metadata/datamodel`

Rerun the `uploadrpd` command. The command will generate a file into the newly created folder and continue with the upload process.

Removing Existing Computers

Remove a failed or redundant computer from a BI Cluster.

Assumptions

- No binaries, configuration or state is deleted from the removed computer.
- Cluster Controller, Scheduler and BI Server mastership is unchanged.
- You cannot remove the master computer.
- Service Instance registrations can be added or removed.
- The BI system can be running (online) or stopped (offline).
- You must have appropriate file system (offline) or Weblogic Administrator (online) permissions.
- The command does not result in a loss of service.
- The command can only result in a loss of availability if forced by the user.

Prerequisites

If possible, you should stop active components on the target computer before removing it using the `status.sh` and `stop.sh` scripts. See [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#).

1. 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, <machine>)
```

Or run the `delete_bi_machine.sh` script in:

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

2. The command displays the removed computer name.

Post Conditions

- Computer, Managed Server, Node Manager and System Components are removed from that computer.
- Service Instances are unregistered from that computer.
- Ports are unallocated.

Validating That Your System Has Been Scaled Correctly

Components need to be validated to ensure they are configured properly for your system's size and scope.

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

This section contains the following topics:

- [Using Fusion Middleware Control to View System Component Availability](#)
- [Using the Administration Console to View Managed Server Availability](#)

For information about using a command to see the status, see [Viewing the Status of Oracle Business Intelligence Components in a Domain](#).

Using Fusion Middleware Control to View System Component Availability

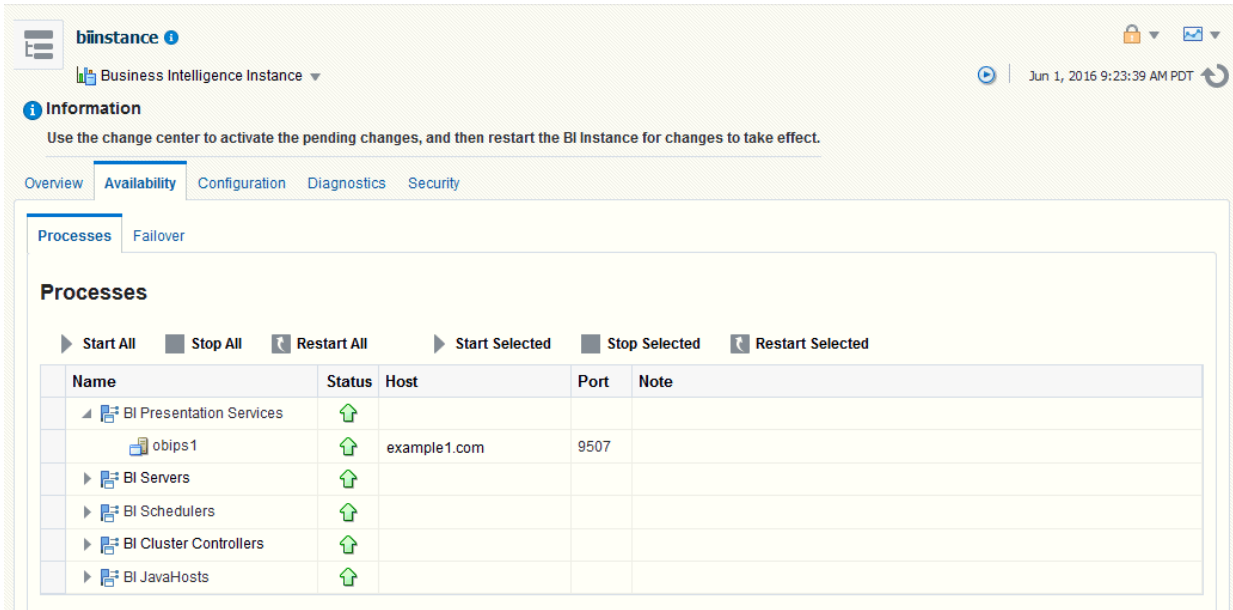
You can use Fusion Middleware Control to view the status of all system components in your deployment.

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence Pages in Fusion Middleware Control](#).
2. 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.

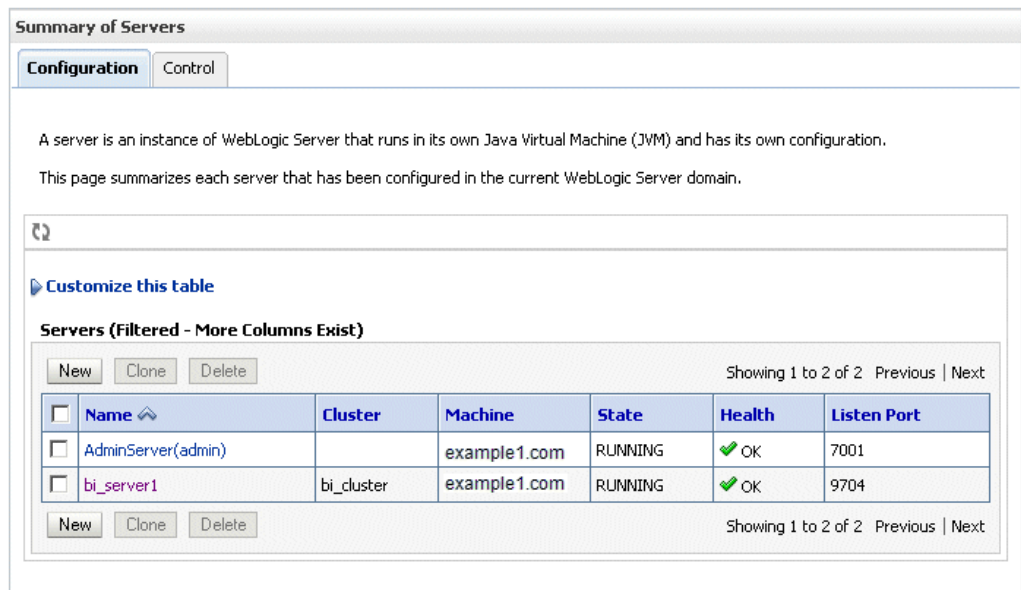


Using the Administration Console to View Managed Server Availability

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.



4

Deploying Oracle Business Intelligence for High Availability

This chapter describes how to configure Oracle Business Intelligence 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 Administration Tool.

This chapter does not 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 Oracle Business Intelligence deployments, see the following documents:

- [Deploying High Availability for Other Components in *High Availability Guide*](#) explains how to implement high availability across the stack, including how to configure a fault tolerant HTTP load balancer and a highly available database for the Oracle Business Intelligence schemas
- [Enterprise Deployment Guide for Oracle Business Intelligence](#) explains how to deploy Oracle Business Intelligence 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

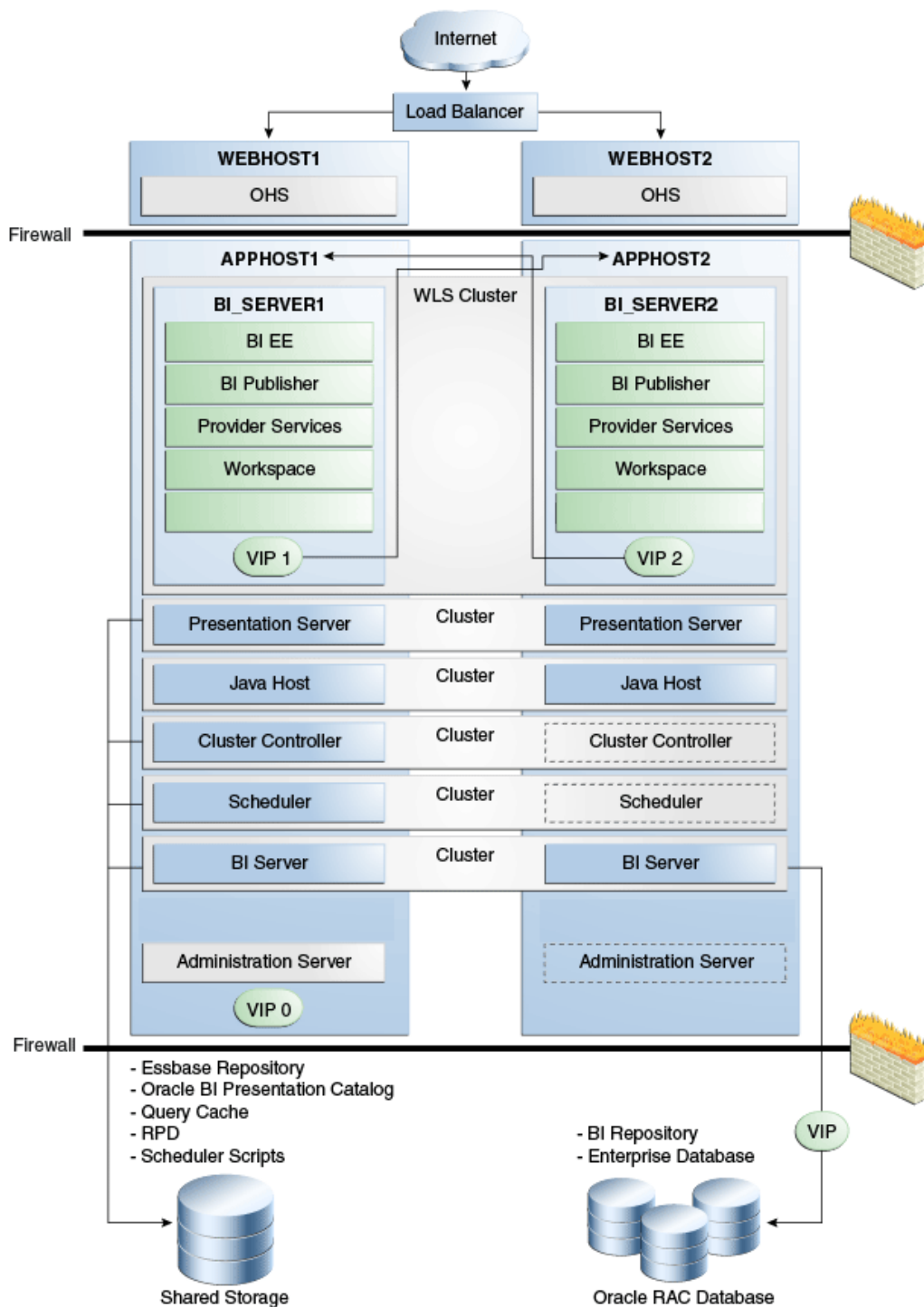
This chapter includes the following sections:

- [About Oracle Business Intelligence Components in a Clustered Environment](#)
- [Configuring Oracle Business Intelligence Components for High Availability](#)
- [Optional Configuration for Oracle Business Intelligence High Availability](#)
- [Using the Cluster Manager](#)
- [Troubleshooting a Fusion Middleware Control Clustered Environment](#)

About Oracle Business Intelligence Components in a Clustered Environment

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

See [About the Administration Server, Managed Servers, and System Components](#) for more information about system components and Java components.



In the figure above, the Oracle Business Intelligence 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.

Oracle BI Presentation Services, JavaHost, Oracle BI Cluster Controller, Oracle BI Scheduler, and Oracle BI 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 are started but do not service requests) and are only made active if APPHOST1 components fail.

Customer metadata is stored in the shared SDD (as BAR files for import or export).

Recommendations for Availability

In a production system, it is 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 [Using 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 *Configuring High Availability for Oracle Business Intelligence and EPM 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 are not supported.
- The clock on each server participating in a cluster must be kept in synchronization. Out-of-sync clocks can skew reporting.

Using Fusion Middleware Control to Identify Single Points of Failure

If there is 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 are familiar with the information in [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence 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.

Achieving High Availability Using an Active-Passive Model

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

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

A two-node Cold Failover Cluster can be used to achieve active-passive availability for Oracle Business Intelligence. 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 Business Intelligence continues servicing clients from that node. All Oracle Business Intelligence components are failed over to the new active node. No Oracle Business Intelligence components run on the failed node after the failover.

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

Configuring Oracle Business Intelligence Components for High Availability

To configure Oracle Business Intelligence 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 have 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 for Oracle Business Intelligence.

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

Optional Configuration for Oracle Business Intelligence High Availability

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

This section contains the following topics:

- [Setting Optional Cluster Controller Parameters](#)
- [Setting Optional Presentation Services Parameters](#)
- [Setting Optional Oracle BI Presentation Services Plug-in Parameters](#)

Setting 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

- 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.

Parameter	Description	Default Value
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.	5
ControllerPollSeconds	The frequency in seconds of heartbeat messages between the Cluster Controllers.	5

- Save and close the file.
- Restart Oracle Business Intelligence.

This code shows example parameters in the `bi_cluster_config.xml` file. Note that any additional elements that are not shown in this example are centrally managed and cannot be set manually.

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

Setting 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.

- Open the configuration file `instanceconfig.xml` for editing at:

BI_DOMAIN/config/fmwconfig/biconfig/OBIPS

- Under the `ServerInstance` tag, the `JavaHostProxy` element has optional subelements. The following table describes these optional subelements.

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.

- Save and close the file.
- Restart Oracle Business Intelligence.

Setting Optional Oracle BI Presentation Services Plug-in Parameters

You can optionally configure the Oracle BI 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/bridgeconfig.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 Oracle BI Presentation Services Plug-in component score is too low.

The instance score is an internal score that the load balancing algorithm associates with each Oracle BI Presentation Services Plug-in instance in the cluster. It is 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 Oracle BI Presentation Services Plug-in is using the Oracle BI Presentation Catalog, then the **xmlpserver** application must also be restarted.

Using the Cluster Manager

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

The Cluster Manager is still supported in the current release.

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 Master BI Server. Fusion Middleware Control shows the current status of clustered components, but does not provide a way to view the current state.

The Cluster Manager lets you 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 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 cannot 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 Administration Tool, open a repository in online mode.
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.

Viewing and Managing 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 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 is hosting the Oracle BI Server or Cluster Controller.
Role	The role of the object in the cluster: <ul style="list-style-type: none"> • Controlling. A Cluster Controller that is currently assigned the responsibility for control of the cluster. • Primary. The Primary Cluster Controller. This role is not displayed if the Primary Cluster Controller is currently the controlling Cluster Controller. • Secondary. The Secondary Cluster Controller. This role is not displayed if the Secondary Cluster Controller is currently the controlling Cluster Controller. • Clustered server. An Oracle BI Server that is a member of the cluster. This role is not displayed for the clustered server that is defined as the master server. • Master. The clustered server that the Administration Tool connects to for editing repositories in online mode. • 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.

Column	Description
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.
Status	<p>The status of the object in the cluster:</p> <ul style="list-style-type: none"> • 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 cannot accept session requests or assign sessions to available servers within the cluster. Offline Oracle BI Servers do not communicate with the controlling Cluster Controller and cannot 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 is not assigned requests. The server remains in this state until the Start command is issued against this server from the 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 executes 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	<p>When Clusters is selected in the Explorer pane, this field is available. There are three types:</p> <ul style="list-style-type: none"> • 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 is associated with the cache entry.
Column count	Number of columns in each row of this cache entry's result set.

Column	Description
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.
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 is, 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 next 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 is 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 is 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 is 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.
Start Time	Time of the initial query request.
Status	<p>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.</p> <ul style="list-style-type: none"> • Idle. There is presently no activity on the request or session. • Fetching. The request is being retrieved. • Fetches. 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 is 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.

Troubleshooting a Fusion Middleware Control Clustered Environment

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

See [Using Fusion Middleware Control to View System Component Availability](#) and [Using the Administration Console to View Managed Server Availability](#).

After enabling 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 are not 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 or UNIX.

See [Diagnosing and Resolving Issues in Oracle Business Intelligence](#) .

Avoiding Errors with Network Appliance Devices When the Oracle BI Server Is Running on Linux or UNIX

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

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

Linux or UNIX 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 UNIX permissions on NTFS qtrees after the design file is saved:

```
options cifs.ntfs_ignore_unix_security_ops on
```

5

Managing Performance Tuning and Query Caching

This chapter describes ways to improve Oracle Business Intelligence 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

This chapter includes the following sections:

- [Monitoring Service Levels](#)
- [About Query Performance Tuning](#)
- [Setting Performance Parameters in Fusion Middleware Control](#)
- [About the Oracle BI Server Query Cache](#)
- [Configuring Query Caching](#)
- [Monitoring and Managing the Cache](#)
- [Strategies for Using the Cache](#)
- [Cache Event Processing with an Event Polling Table](#)
- [Configuring the Data Source Cache](#)
- [Managing the Oracle BI Presentation Services Cache Settings](#)
- [Improving Oracle BI Web Client Performance](#)
- [Setting the JVM Heap Size for Oracle Business Intelligence](#)
- [Improving Performance when Downloading Large Reports](#)
- [Capturing Metrics Using the Dynamic Monitoring Service](#)

Monitoring Service Levels

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

Oracle Business Intelligence automatically and continuously measures runtime performance in real time. The performance metrics are automatically enabled; you do not 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 Business Intelligence installation. If you encounter a problem, such as an application that is 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 WSLT Commands in WSLT 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:

- [Using Fusion Middleware Control to View All Oracle Business Intelligence Metrics](#)
- [Using the Administration Console to View Metrics for Java Components](#)

Using Fusion Middleware Control to View All Oracle Business Intelligence Metrics

You can view and graph all the available Oracle Business Intelligence metrics from the Performance Summary page 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 Oracle Business Intelligence 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.

Using 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**. The Summary of Servers page is displayed.
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 [Managing the Query Log](#).

- **Aggregate tables:** It is extremely important to use aggregate tables to improve query performance. Aggregate tables contain precalculated summarizations of data. It is 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 have been properly specified in the repository. See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* 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 does not improve performance for most ad-hoc analysis. See [About the Oracle 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 [Setting 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 [Scaling Your Deployment](#).

Setting 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:

- [Using Fusion Middleware Control to Disallow RPD Updates](#)
- [Using Fusion Middleware Control to Set the User Session Log-Off Period](#)
- [Using Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables](#)
- [Using Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View](#)

Using Fusion Middleware Control to Disallow RPD Updates

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

Setting this parameter affects whether you can update the repository when the Administration Tool connects in both online and offline mode. It also affects whether you can perform other repository update operations using other utilities, such as biserverxmlcli. Note that the aggregate persistence feature is not available when repository updates are prevented.

Preventing repository updates can improve Oracle BI Server performance, because in this mode the Oracle BI Server does not need to handle lock control.

If you choose to prevent repository updates, then when the Administration Tool opens a repository in either online or offline mode, a message informs the user that the repository is read-only.

Before you begin this procedure, ensure that you are familiar with the information in [Using 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 repository 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 [Mapping User Interface Labels with Configuration File Elements](#).

Using 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 are familiar with the information in [Using 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 execute 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 [Mapping User Interface Labels with Configuration File Elements](#).

Using 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 [Configuring for Displaying and Processing Data in Views](#).

Before you begin this procedure, ensure that you are familiar with the information in [Using 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 [Using 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 [Mapping User Interface Labels with Configuration File Elements](#).

Using 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 Oracle BI Server for rendering a view.

Reducing the number of rows in a table can significantly improve performance by reducing the system resources that can be consumed by a given user session.

Advanced configuration settings are described in [Configuring for Displaying and Processing Data in Views](#).

Before you begin this procedure, ensure that you are familiar with the information in [Using 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 Business Intelligence 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 [Using 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 [Mapping 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 Configuring Cube Settings for Pivot Tables and Graphs](#)).

For information about corresponding configuration file elements, see [Mapping User Interface Labels with Configuration File Elements](#).

About the Oracle BI Server Query Cache

You can configure the Oracle BI Server to maintain a local, disk-based cache of query result sets (query cache).

The query cache enables the Oracle 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 Oracle BI 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 [Using the Cache Manager](#).
- **Automatically.** In the Administration Tool, you can disable cache for the system, set caching attributes for a specific physical table, and use Oracle Business Intelligence event tables to purge cache automatically. See [Monitoring and Managing the Cache](#).
- **Programmatically.** The Oracle 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 [Purging and Maintaining 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 [Using Agents to Seed the Oracle BI Server 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 Oracle 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 UNIX 64-bit architectures. Incompatible cache entries are automatically removed.

Note that query cache entries are *not* portable across different releases of Oracle Business Intelligence, such as between Version 10.1.3.2 and 11.1.1.

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, the 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 the intermediate results do not have to come over the network to the 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 it could save money in the budget as well.

Another benefit of using the cache to answer a query is savings in processing time on the 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 has the following advantages:

- Dramatic improvement of query performance
- Less network traffic
- Reduction in database processing
- Reduction in Oracle BI Server 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.

The following sections discuss the costs of caching.

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

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.

Keeping the Cache Up To Date

If the cache entries are not 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 are performing yearly summaries of the different divisions in the company. New data does not 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 are 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 data mart 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 data mart, 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 data sets.

Cache Sharing Across Users

If shared logon has been enabled for a particular connection pool, then the cache can be shared across users and does not need to be seeded for each user.

If shared logon has not been enabled and a user-specific database login is used, then each user generates their own cache entry.

See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* 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 is not 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 *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

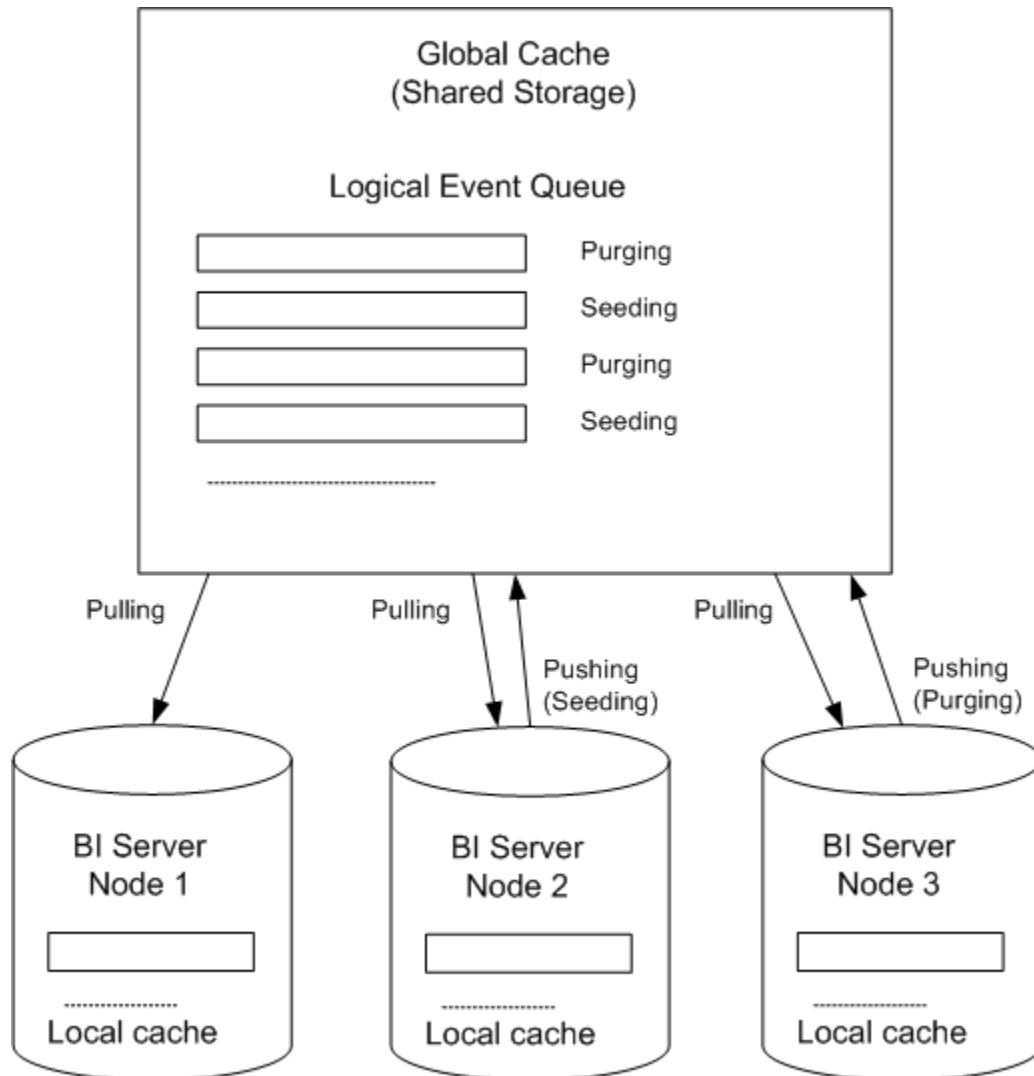
About the Global Cache

In a clustered environment, Oracle BI 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 Oracle BI Presentation Services nodes push to and pull from the logical event queue. Each Oracle BI 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 Oracle BI 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 Oracle BI Presentation Services Node 2 pushing a seeding event to the queue and Oracle BI Presentation Services Node 3

pushing a purging event to the queue. The arrows from the shared storage to each Oracle BI Presentation Services node show each node pulling from the common location. This occurs on a periodic basis and enables participating Oracle BI Presentation Services nodes to obtain updates to the logical event queue made by other Oracle BI Presentation Services.



The Oracle BI 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 Oracle BI Presentation Services node locks the logical event queue on the shared storage and then pushes in the seeding or purging event. If there is 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 Oracle BI Presentation Services nodes. The queue is sorted according to the timestamp of the events. Hence, clocks on all Oracle BI Presentation Services nodes participating in cluster must be synchronized.

Each Oracle BI 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 Oracle BI Presentation Services nodes that participate in the cluster does not 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 BI 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 Oracle BI Presentation Services node that participates in the cluster. See [Using Fusion Middleware Control to Set Global Cache Parameters](#) and [Manually Editing Additional Global Cache Parameters](#).

A seeding or purging procedure is submitted to a specific Oracle BI Presentation Services node. If that Oracle BI Presentation Services is a node in a BI cluster and the global cache parameters have been defined in Oracle BI Presentation Services configuration files, then the seeding or purging events are propagated across all Oracle BI Presentation Services nodes that participate in the same clustered environment.

Configuring 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 also must decide on a strategy for flushing outdated cache entries; see [Monitoring and Managing the Cache](#).

This section contains the following topics:

- [Using Fusion Middleware Control to Enable and Disable Query Caching](#)
- [Using Fusion Middleware Control to Set Query Cache Parameters](#)
- [Manually Editing Additional Query Cache Parameters](#)
- [Using Fusion Middleware Control to Set Global Cache Parameters](#)
- [Manually Editing Additional Global Cache Parameters](#)

Using 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 are familiar with the information in [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence 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 [Mapping User Interface Labels with Configuration File Elements](#).

Using 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 are familiar with the information in [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence 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 [Mapping User Interface Labels with Configuration File Elements](#).

Manually Editing 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 is not cached.
- Typically, if a query gets a cache hit from a previously executed query, then the new query is not 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 executed query.

See [Configuration File Settings](#) for more information about the additional query cache parameters.

Using 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 are familiar with the information in [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence 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 [Mapping User Interface Labels with Configuration File Elements](#).

Manually Editing 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 `nquery.log`. You can find this file at:

`BI_DOMAIN/servers/obisn/logs`

See [Configuration File Settings](#).

Monitoring and Managing 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 have changed, and a process to monitor, identify, and remove any undesirable cache entries.

This section contains the following topics:

- [Choosing a Cache Management Strategy](#)
- [Purging and Maintaining Cache Using ODBC Procedures](#)
- [How Repository Changes Affect the Query Cache](#)

Choosing 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.

Disabling 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 are 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 is associated with a particular business model before enabling the cache again.

See [Using Fusion Middleware Control to Enable and Disable Query Caching](#).

Caching 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 might not be good candidates to include in the cache unless you use the Cache Persistence Time settings. For example, suppose that you have a table that stores stock ticker data that is updated every minute. You could use the Cache Persistence Time settings to purge the entries for that table every 59 seconds.

You can also use the **Cache persistence time** field 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 the 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 do not want cache entries to automatically expire, select **Cache never expires**.
4. Click **OK**.

Configuring Oracle BI Server Event Polling Tables

Oracle BI Server event polling tables store information about updates in the underlying databases.

An application (such as one that loads data into a data mart) could be configured to add rows to an event polling table each time a database table is updated. The Oracle 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 [Setting Up Event Polling Tables on the Physical Databases](#).

Purging and Maintaining 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 repository 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 repository 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 Oracle Business Intelligence repository, run the following call in a script:

```
Call SAPurgeCacheByTable( 'DBName', 'CatName', 'SchName', 'TabName' );
```

Note:

Wildcards are not supported by the Oracle BI Server for this function. In addition, `DBName` and `TabName` cannot be null. If either one is null, then an error message is displayed.

- **SAPurgeAllCache.** Purges all cache entries. The following is an example of this call:

```
Call SAPurgeAllCache();
```

- **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 cannot be null. The following shows the syntax of this call:

```
Call SAPurgeCacheByDatabase( 'DBName' );
```

About ODBC Procedure Syntax

If there is 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 are not 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 has been selected in the repository physical database object
- Whether any session variables have been marked as **Security Sensitive** in the repository

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 [Managing the Oracle BI 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 is not enabled.
59116	The database specified does not exist.
59117	The table specified does not exist.

Storing and Purging Cache for SAP/BW Data Sources

Due to differences in naming conventions between Microsoft Analysis Services and SAP/BW data sources, there is a cache subsystem for storing member unique names.

In Microsoft Analysis Services, member caption name is the same as member unique name. However, in SAP/BW data sources, member caption name is different from member unique name. Therefore, the Oracle BI Server maintains a cache subsystem for SAP/BW member unique names. This subsystem is turned off by default. For configuration information, see the topic about the MDX Member Name Cache Section in [Configuration File Settings](#).

When a query is received for member unique name, the subsystem checks the cache to determine whether cache exists for this query. If cache exists, then the record for the cached unique name is returned. If there is no cache that matches the query, then the subsystem sends a probing query to SAP/BW.

The probing query is logged when the log level is equal or greater than 2. The status of the subsystem, such as if the subsystem is enabled and events such as start and shutdown events, are also written to the server log.

Caution:

With each increased logging level, performance is impacted. Use caution when increasing the log level for users.

Be aware of the following cache purge issues:

- The size of multidimensional cache entries can grow very large. Therefore, a limit on the size of each member set has been established in the `MDX_MEMBER_CACHE` section of the `NQSConfig.INI` file.
- The format of persisted cache might not be consistent after an upgrade. Therefore, purge all cache before a software upgrade.
- The cache is populated the first time that the query runs. Arrange to populate the cache during off-peak hours, to minimize performance impact.

Note:

In the Administration Tool, you can purge cache for an individual cube table by right-clicking the cube table, and then selecting **Purge Member Cache**. This must be performed in online mode by a user with administrator privileges.

The following purge procedures are specific to SAP/BW data sources:

- **SAPurgeALLMCNCache.** Purges all SAP/BW cache entries.

The following shows the syntax of this procedure:

```
SAPurgeALLMCNCache ( )
```

- **SAPurgeMCNCacheByCube.** Purges all cache entries that are associated with the specified physical cube. The database name and cube name are the external names of the repository objects. The following shows the syntax of this procedure:

```
SAPurgeMCNCacheByCube( 'DBName', 'CubeName' )
```

The next table describes the messages that are returned.

Return Code	Return Message
1	SAPurgeALLMCNCache returns successfully.
1	SAPurgeMCNCacheByCube returns successfully.

Return Code	Return Message
59116	The database specified does not exist. If the database and physical cube are both wrong, then this result code is returned.
85025	The physical cube specified does not exist.

Only users with administrative privileges can run ODBC purge procedures.

How Repository Changes Affect the Query Cache

When you modify Oracle Business Intelligence repositories, 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 repository 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 three scenarios to be aware of: when the changes occur in online mode, when they occur in offline mode, and when you are switching between repositories.

Online Mode

When you modify an Oracle Business Intelligence repository in online mode, 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 repository, 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 layer purge all cache entries for that business model.

Offline Mode

When you modify an Oracle Business Intelligence repository in offline mode, you might make changes that affect queries that are stored in the cache and render those cached results obsolete. Because the repository is not loaded by the server during offline mode edits, the server has no way of determining if the changes made affect any cached entries. The server therefore does not automatically purge the cache after offline changes. If you do not purge the cache, then there might be invalid entries when the repository is next loaded. Unless you are sure that there are no entries in the cache that are affected by your offline changes, then purge the cache for any business model that you have modified.

Switching Between Repositories

If you intend to remove a repository from the configuration of the Oracle BI Server, then ensure that you purge the cache of all cache entries that reference the repository. Failure to do so results in a corrupted cache. See [Purging Cache in the Administration Tool](#).

Changes to Dynamic Repository Variables

The values of dynamic repository variables are refreshed by data that is returned from queries. When you define a dynamic repository variable, you create an initialization block or use a preexisting one that contains a SQL query. You also configure a schedule for the Oracle BI Server to follow to execute the query and periodically refresh the value of the variable.

If the value of a dynamic repository variable changes, then any BI Server 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 is not removed immediately, but remains until it is cleaned through the usual caching mechanism.

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](#)
- [Running a Suite of Queries to Populate the Cache](#)
- [Using Agents to Seed the Oracle BI Server Cache](#)
- [Using the Cache Manager](#)

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 the server was able to use cache to answer the query and did not go to the database at all. The 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 <code>SELECT</code> list must match	<p>All of the columns in the <code>SELECT</code> 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.</p> <p>This rule describes the minimum requirement to hit the cache, but meeting this rule does not guarantee a cache hit. The other rules listed in this table also apply.</p>

Factor or Rule	Description
Columns in the <code>SELECT</code> list can be composed of expressions on the columns of the cached queries	<p>The 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:</p> <pre>SELECT product, month, averageprice FROM sales WHERE year = 2000</pre> <p>hits cache on the query:</p> <pre>SELECT product, month, dollars, unitsales FROM sales WHERE year = 2000</pre> <p>because <code>averageprice</code> can be computed from <code>dollars</code> and <code>unitsales</code> (<code>averageprice = dollars/unitsales</code>).</p>
<code>WHERE</code> clause must be semantically the same or a logical subset	<p>For the query to qualify as a cache hit, the <code>WHERE</code> clause constraints must be either equivalent to the cached results, or a subset of the cached results.</p> <p>A <code>WHERE</code> clause that is a logical subset of a cached query qualifies for a cache hit if the subset meets one of the following criterion:</p> <ul style="list-style-type: none"> A subset of <code>IN</code> list values. Queries requesting fewer elements of an <code>IN</code> list cached query qualify for a cache hit. For example, the following query: <pre>SELECT employeename, region FROM employee, geography WHERE region in ('EAST', 'WEST')</pre> <p>qualifies as a hit on the following cached query:</p> <pre>SELECT employeename, region FROM employee, geography WHERE region in ('NORTH', 'SOUTH', 'EAST', 'WEST')</pre> It contains fewer (but identical) <code>OR</code> constraints than the cached result. It contains a logical subset of a literal comparison. For example, the following predicate: <pre>WHERE revenue < 1000</pre> <p>qualifies as a cache hit on a comparable query with the predicate:</p> <pre>WHERE revenue < 5000</pre> There is no <code>WHERE</code> clause. If a query with no <code>WHERE</code> clause is cached, then queries that satisfy all other cache hit rules qualify as cache hits regardless of their <code>WHERE</code> clause. <p>In addition columns that are used on the <code>WHERE</code> clause must be on the projection list. For example, the following query:</p> <pre>SELECT employeename FROM employee, geography WHERE region in ('EAST', 'WEST')</pre> <p>Does not result in a cache hit for the seeding query in the previous list because <code>REGION</code> is not on the projection list.</p>

Factor or Rule	Description
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.
Queries with special functions must be an exact match	Other queries that contain special functions such as time series functions (<code>AGO</code> , <code>TODATE</code> , and <code>PERIODROLLING</code>), limit and offset functions (<code>OFFSET</code> and <code>FETCH</code>), relationship functions (<code>ISANCESTOR</code> , <code>ISLEAF</code> , <code>ISROOT</code> , and <code>ISSIBLING</code>), 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 <code>WHERE</code> 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, <code>SELECT * FROM product</code> does not match <code>SELECT * FROM product, sales</code> .
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 is not hit. In addition, the value of session variables that are security sensitive must match the security session variable values that are defined in the repository, even though the logical SQL statement itself does not reference session variables. See Ensuring 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.
<code>DISTINCT</code> attribute must be the same	If a cached query eliminates duplicate records with <code>DISTINCT</code> processing (for example, <code>SELECT DISTINCT...</code>), then requests for the cached columns must also include the <code>DISTINCT</code> processing; a request for the same column without the <code>DISTINCT</code> 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: <pre>SELECT supplier, region, city, qtysold FROM suppliercity</pre> The following query requests the quantity sold at the city level: <pre>SELECT city, qtysold FROM suppliercity</pre> The second query results in a cache hit on the first query.
Limited additional aggregation	For example, if a query with the column <code>qtysold</code> is cached, then a request for <code>RANK(qtysold)</code> results in a cache miss. Additionally, a query that requests <code>qtysold</code> at the country level can get a cache hit from a query that requests <code>qtysold</code> at the country, region level.

Factor or Rule	Description
ORDER BY clause must be comprised of columns in the select list	Queries that order by columns that are not 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 <code>USE_ADVANCED_HIT_DETECTION</code> to <code>YES</code> in the <code>NQSConfig.INI</code> file. Advanced hit detection enables an expanded search of the cache for hits. See USE_ADVANCED_HIT_DETECTION .
Diagnosing cache hit behavior	To better assess cache hit behavior, set the <code>ENABLE_CACHE_DIAGNOSTICS</code> session variable to 4, as shown in the following example: <code>ENABLE_CACHE_DIAGNOSTICS=4</code>

Ensuring 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, the 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 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 are 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 do not match.
- Session Variable object.** For variables that you are using for authentication, 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 *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*
- Managing Session Variables in *Security Guide for Oracle Business Intelligence Enterprise Edition*
- Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for general information about database and session variable objects

Running 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 does not alleviate.

When the Oracle BI Server refreshes repository variables, it examines business models to determine if they reference those repository variables. If they do, the Oracle BI Server then purges all cache for those business models. See [Changes to Dynamic Repository Variables](#).

Using Agents to Seed the Oracle BI Server Cache

You can configure agents to seed the Oracle BI Server 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 execute requests that refresh this data.

1. Log in to Oracle Business Intelligence 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. Optionally, 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 BI 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 do not 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 do not address ad-hoc queries or drills.

Using 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 repository.

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 Administration Tool toolbar, select **Manage**, then select **Cache**.
2. Click the **Cache** tab to view the cache entries for the current repository, 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.

You can control the cache entry information and its display sequence using the Options settings, select **Edit**, then select **Options** from the Cache Manager, or select **Tools**, then **Options**, then **Cache Manager** from the Administration Tool menu. Information can include the options that are described in the next table.

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 is older than the true value.)
Creation elapsed time	The time, in seconds, that is needed to create the result set for this cache entry. The value that is 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.

Option	Description
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 is 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 is associated with the cache entry.
Repository	The name of the Oracle Business Intelligence repository that is associated with this cache entry.
SQL	The SQL statement that is associated with this cache entry. If subrequests are being cached, then there might be multiple cache entries that are associated with a single SQL statement.
Query Server	The Oracle BI Server that serviced the query.
Fact Table Source	The fact table that is associated with the logical request for this cache entry.

Expand the repository 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.

Displaying 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 <code>MAX_CACHE_ENTRIES</code> parameter in the <code>NQSConfig.INI</code> 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 <code>MAX_ROWS_PER_CACHE_ENTRY</code> parameter in the <code>NQSConfig.INI</code> file.
Number of entries currently in cache	The current number of entries in the global cache. These entries might relate to multiple repositories.

Column	Description
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 repository that you have open and the current global cache information. If the DSN is clustered, then information about all repositories in the cluster is displayed.

Purging Cache in the Administration Tool

Purging cache is the process of deleting entries from the query cache.

You can purge cache entries in the following ways:

- Manually, using the Administration Tool Cache Manager facility (in online mode).
- 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 [Purging and Maintaining Cache Using ODBC Procedures](#).

In addition, cache can be purged when the value of dynamic repository variables changes. See [Changes to Dynamic Repository Variables](#).

You can manually purge cache entries in the Cache Manager as follows:

1. Use the Administration Tool to open a repository in online mode.
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 repository.
- 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 is 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 [Setting Up Event Polling Tables on the Physical Databases](#). It is normally exposed only in the Physical layer of the Administration Tool, where it is 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 it 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 is not completely up to date, there is always the potential for stale data in the cache. See [Populating 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 Business Intelligence repository, 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:

- [Setting Up Event Polling Tables on the Physical Databases](#)
- [Making the Event Polling Table Active](#)
- [Populating the Oracle BI Server Event Polling Table](#)
- [Troubleshooting Problems with Event Polling Tables](#)

Setting 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 Administration Tool. Then, ensure that the feature `PERF_PREFER_IN_LISTS` is not 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 Business Intelligence Platform (BIPLATFORM) schemas in your physical database. RCU creates an event polling table called `S_NQ_EPT`. See *Installing and Configuring Oracle Business Intelligence* for information about running the Repository Creation Utility.

Event tables must have the structure that is 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 does not reside in the same database as the physical tables that were updated. Otherwise, set it to the null value.
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 is defined in the Physical layer of the Administration Tool. For example, if the physical database name is 11308Production, and the database name that represents it in the 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 does not 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.

Event Table Column	Data Type	Description
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 does not 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 is defined for the table in the Physical layer of the Administration Tool. Values cannot 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 is 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 <code>DEFAULT CURRENT_TIMESTAMP</code> for Oracle Database. Values cannot be null. Because this column must be a unique value that increases for each row that is 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 <code>FRACTIONAL_SECOND_PRECISION</code> 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 <code>FRACTIONAL_SECOND_PRECISION</code> . For example, for Oracle Database or Teradata, you might want to change <code>FRACTIONAL_SECOND_PRECISION</code> from 0 to 6.
UPDATE_TYPE	NUMBER	Specify a value of 1 in the update script to indicate a standard update. You cannot 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 is 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 Oracle Business Intelligence deployment, a single event polling table is shared by every Oracle BI Server node in the cluster. However, a single event polling table cannot 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 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.

Making 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 Administration Tool, open the repository 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 [Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition](#) 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 is specified for the event table has read and write access to the event table. The repository both reads the table and deletes rows from it, so it needs write permission. Event tables do not need to be exposed in the Business Model and Mapping 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. Do not 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 is no reason to cache results from an event polling table.

Populating the Oracle BI Server Event Polling Table

The Oracle BI Server does not populate the event polling table. The event table is populated by inserting rows into it each time that 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 is not 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.

Troubleshooting 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 `nqserver.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 `nqserver.log` and the `nqquery.log` in the following location:

```
BI_DOMAIN/servers/obisn/logs
```

Configuring the Data Source Cache

To improve performance for data blending, you should configure a data source cache. This can be done 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](#)

Creating 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 is 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.

To 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 will be 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 will be persisted.
# This is only used if file-based XSA cache is configured.
# If a relative path is specified, it will be 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 will be
# persisted.
# If a relative path is specified, it will be 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. 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] [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.

Creating 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 is 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 from 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.](#)

Configuring the Data Source Cache on a Different Database than the Data Warehouse

1. Create the physical objects that represent the cache database in your repository. 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 will be 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 will be
persisted.
# If a relative path is specified, it will be relative to:
# BI_DOMAIN/servers/obisn
DESCRIPTOR_STORAGE_PATH = "xsacache"; # location where the cache metadata files
are stored

```

- Restart BI Server.
- 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.
[2017-01-13T14:41:30.716-07:00] [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 will store the data for the data source in the database. However it will 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, 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 executed internally by BI Server.

- Create the physical objects that represent the cache database in your repository 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 is used for data warehouse queries.

2. Update parameters under the `XSA_CACHE` section in `NQSCConfig.INI` to point to the right schema and connection pool.

```
[XSA_CACHE]
ENABLE = YES;

# The schema and connection pool where the XSA data will be 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 will be
persisted.
# If a relative path is specified, it will be 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 will store the data for uploaded files in the database. However it will store the metadata on the local file system in the directory specified by the `DESCRIPTOR_STORAGE_PATH` parameter under the `XSA_CACHE` section in `NQSCConfig.INI`. To improve performance, we recommend that you put your data source cache metadata files on RAMDisk.

Managing the Data Source Cache

Once you have 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 was 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 Cleaning 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 will be left in an inconsistent state and will result 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`.

Managing the Oracle BI 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 are not stored.

The files for the Presentation Services cache have names such as nQS_XXXX_X_XXXXXX.TMP. The files 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/tmp/obis_temp
```

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 is not running.

The Presentation Services cache is different from the cache that is 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 that are 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>100</MaxEntries>
      <MaxExpireMinutes>60</MaxExpireMinutes>
      <MinExpireMinutes>10</MinExpireMinutes>
      <MinUserExpireMinutes>10</MinUserExpireMinutes>
      <RefreshIncludeBIserverCache>>false</RefreshIncludeBIserverCache>
    </Query>
  </Cache>
</ServerInstance>
```

```
</Cache>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

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.	500
MaxExpireMinutes	Specifies the maximum amount of time, in minutes, that an entry in the cache can exist before it is 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 is 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 has 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. Starting in Oracle Business Intelligence 12c, 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 <i>false</i> to revert to the previous behavior.	true

Improving Oracle BI Web Client Performance

You can improve the performance of the Oracle BI 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:

- [Configuring Apache HTTP Server for Static File Caching](#)
- [Configuring Oracle HTTP Server for Static File Caching](#)

Configuring 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
    WLogFile /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.

To 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 is not 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 do not 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. This header is required to force earlier versions of Microsoft Internet Explorer to properly cache static files.

– 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 must not be exposed to web clients.

3. Save and close the file.

Configuring Oracle HTTP Server for Static File Caching

Configuration for Oracle HTTP Server is similar to configuration for Apache HTTP Server, except that you do not 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*.

Setting the JVM Heap Size for Oracle Business Intelligence

You can change the default JVM heap size for the Administration Server and Managed Servers by setting the `USER_MEM_ARGS` parameter in the startup script for Oracle WebLogic Server.

The following procedure sets the same values for both the Administration Server and Managed Servers.

See *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

1. Use the WebLogic Server Administration Console to shut down the servers gracefully.
2. Open the `setDomainEnv.sh` (or `setDomainEnv.bat` on Windows systems) for editing. You can create this file in:

```
BI_DOMAIN/bin
```

3. Set the `-Xmx` argument for `USER_MEM_ARGS`. The following list shows examples of how to set `USER_MEM_ARGS` for various operating systems:

- UNIX shell script (.sh)

```
USER_MEM_ARGS="-Xms256m -Xmx1024m -XX:CompileThreshold=8000 -  
xx:PermSize=128m -XX:MaxPermSize=512m"
```

- Windows command script (.bat)

```
set USER_MEM_ARGS="-Xms256m -Xmx1024m -XX:CompileThreshold=8000 -  
xx:PermSize=128m -XX:MaxPermSize=512m"
```

 **Note:**

The arguments for `USER_MEM_ARGS` can vary, depending on the JDK vendor.

4. After setting the parameter, 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.

Improving Performance when Downloading Large Reports

Downloading large reports in Oracle BI EE 12c 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.

Using 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)

Using 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*.

Capturing Metrics Using the Dynamic Monitoring Service

In addition to the Metrics Browser in Fusion Middleware Control, you can view metrics for Oracle Business Intelligence using the Dynamic Monitoring Service (DMS) and WLST commands.

This section describes how to use these methods.

Using the Dynamic Monitoring Service for Metrics

You can use the Dynamic Monitoring Service (DMS) to view metrics for Oracle Business Intelligence.

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 Business Intelligence. 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 [Using WLST Commands for Metrics..](#)

Using WLST Commands for Metrics

Using WLST, you can collect metrics about the system.

You can use WLST commands to capture metrics for Oracle Business Intelligence.

1. Navigate to the `ORACLE_HOME/oracle_common/common/bin` directory.
2. Run the WLST utility.
3. Connect to the Oracle BI system using the `connect` command, as shown in the following example:

```
connect('user', 'password', 't3://<host><port>')
```

4. Verify that you are in "online mode" by viewing 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','welcome1','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','welcome1','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

Resolving Issues

Learn how to resolve issues in Oracle Business Intelligence.

It includes the following chapters:

- [Diagnosing and Resolving Issues in Oracle Business Intelligence](#)
- [Managing Usage Tracking](#)

6

Diagnosing and Resolving Issues in Oracle Business Intelligence

This chapter describes how to diagnose and resolve issues in Oracle Business Intelligence using tools such as Fusion Middleware Control and log files. This chapter includes the following sections:

- [What Diagnostic Tools Are Available?](#)
- [Collecting Diagnostic Bundles](#)
- [Viewing and Configuring Diagnostic Log Files](#)
- [Understanding Diagnostic Log Files and Log Configuration Files](#)
- [Managing the Query Log](#)
- [Logging in Oracle BI Server](#)
- [Using ODBC/JDBC Procedures to Obtain Oracle BI Server Diagnostics](#)
- [Troubleshooting System Startup](#)

What Diagnostic Tools Are Available?

There are several diagnostic tools available to help you troubleshoot issues.

Oracle Business Intelligence provides various diagnostic tools to assist you in finding 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	Collecting Diagnostic Bundles
Overview page in Fusion Middleware Control	Enables you to view recent issues with the system.	Displaying Oracle Business Intelligence Pages in Fusion Middleware Control
Performance metrics	Enables you to view metrics that affect performance.	Monitoring Service Levels
Diagnostic pages in Fusion Middleware Control	Enables you to drill into problems and view and configure log files.	Using Fusion Middleware Control to View Log Information, Error Messages, and Alerts Using Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels

Tool	Description	Reference
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.	About Usage Tracking
Reports of Catalog objects	Enables you to learn details of objects in the Oracle BI Presentation Catalog.	Creating Reports to Display Catalog Data Using Catalog Manager
Consistency Check Manager	Enables you to check the validity of the repository.	Checking the Consistency of a Repository or a Business Model in <i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>
Model Check Manager	Enables you to check for modeling problems that might affect Oracle BI Summary Advisor and the aggregate persistence engine.	"Using Model Check Manager to Check for Modeling Problems" in <i>Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition</i>
ODBC/JDBC procedures	Enables you to obtain diagnostic information for the Oracle BI Server.	Using ODBC/JDBC Procedures to Obtain Oracle BI Server Diagnostics

Collecting Diagnostic Bundles

This section explains how to collect diagnostic bundles needed by Oracle Support to help resolve issues.

You must collect post configuration diagnostic bundles before Oracle Development or Support will accept your bug or support request (SR) submissions.

Assumptions:

- You must review the product FAQ.
- BI Installation and configuration was successful, specifically the BI 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. If the BI Configuration Assistant fails see Running the Oracle Business Intelligence 12c Configuration Assistant *Installing and Configuring Oracle Business Intelligence*.
- 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.

Viewing and Configuring 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:

- [Using Fusion Middleware Control to View Log Information, Error Messages, and Alerts](#)
- [Configuring Log File Rotation Policy and Specifying Log Levels](#)

Using Fusion Middleware Control to View Log Information, Error Messages, and Alerts

You can search for and view the log entries for Oracle Business Intelligence components using Fusion Middleware Control Log Viewer.

The log files can be searched 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 is 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) do not 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 [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence 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.

Configuring Log File Rotation Policy and Specifying 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:

- [Using Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels](#)
- [Manually Changing Additional Log File Settings](#)

Using 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 are familiar with the information in [Using 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 changes to be made.
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 Changing 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 Business Intelligence, 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 Business Intelligence.

Diagnosing 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 Oracle Business Intelligence 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 are interested.

3. To start filtering the list, enter search criteria to locate the messages in which you are 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 is 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 Business Intelligence system was doing, which can assist you in determining where something went wrong.

4. To filter for the messages for Oracle Business Intelligence
 - 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 for Oracle Business Intelligence 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 are 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 Oracle Business Intelligence was doing. For example, a value of nqserver.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 repository, but you cannot see the repository 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 repository was copied has run out of memory. An earlier error message indicates that the Administration Server had reported the change to the repository and had tried to synchronize the change to the Managed Server.

Understanding 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 are files used to store message information that is generated by Oracle Business Intelligence 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 in Oracle Business Intelligence:

- Presentation Services
 - *\CatalogCrawler\sawcatalogcrawlerlogsysn.log* — The catalog crawler log file, which is not searchable in the Fusion Middleware Control Log Viewer.
 - *sawlogn.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 [Logging in Oracle BI Server](#).

- Oracle BI Server
 - *obis<n>_query.log* — The Oracle BI Server query log, which is not searchable in the Fusion Middleware Control Log Viewer.
For example, where *<n>* = -1, -2.
Note: The date and timestamp is in the log file.
 - *nqserver<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 the Oracle BI Administration Tool.
 - Oracle BI Server utilities — For example, *biserverxmlexec* and *equalizerpds*, also generate their own logs when they are 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 Oracle BI 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 cannot set the time zone in which messages are logged in the files: nqcluster.log, nqscheduler.log, and nqserver<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 Business Intelligence.

 **Note:**

Editing a diagnostic log configuration file for a single component is not advised, because changes might subsequently be overwritten.

Log configuration files for Oracle Business Intelligence 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 6-1](#) and [Example 6-2](#) illustrate two of the log configuration files for Oracle Business Intelligence.

Example 6-1 BI Server Diagnostic Log Configuration File Format — Example 1

```
<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 6-2 JavaHost Server Diagnostic Log Configuration File Format — Example 2

```
<?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='C:\oracle_bi_ee_BIFNDNTPSNT0911060426S-Release
\jhlogs\javahost.log' />
      <property name='maxFileSize' value='1000000' />
      <property name='maxLogSize' value='5000000' />
    </log_handler>
  </log_handlers>
  <loggers>
    <logger name='saw' level='NOTIFICATION:1'
useParentHandlers='false'>      <handler name='odl-handler' />
    </logger>
  </loggers>
</logging_configuration>
```

Oracle Business Intelligence components control their diagnostic log files by using server-specific settings in their log configuration files, for example:

- Oracle BI Presentation Services log configuration file:
 - `writerClassId` settings configure messages that the system writes to the `sawlog.log` file.

- Oracle BI Server log configuration file:
 - `Log` settings configure messages that the system writes to the `nqserver.log` file.
See [What Messages Are Included in the System Log?](#)
 - `UserLog` settings configure messages that the system writes to the `nquery.log` file.
See [Managing 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 for Oracle Business Intelligence 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.

Category:Level	Description
Trace:1	<p>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.</p> <p>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.</p>
Trace:16	<p>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.</p> <p>High performance impact. This level must not be enabled on a production environment, except on special situations to debug issues with the software.</p>
Trace:32	<p>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.</p> <p>Very high performance impact. This level is not expected to be enabled in a production environment and developers use it only to debug the software on a test or development environment.</p>

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 [Using 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 is 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 is older than the MaximumLogAgeDay setting is deleted.

Different Oracle BI 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 [Using 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 `nqserver.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 UNIX 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 cannot 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.
```

Managing the Query Log

The Oracle BI Server provides a facility for logging 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. It is a resource-intensive process if you track the entire user community.

Note:

For production systems, it is 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 [Managing Usage Tracking](#).

It is recommended that you test users only when the user name clearly indicates it is a test user and have verified that query logging is enabled. If logging is enabled for such users, then it is 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 should not 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 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:

- [Configuring Query Logging](#)
- [Using the Log Viewer](#)

Configuring 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 repository 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 Oracle BI Administration Tool.

Setting 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 [Setting the Query Logging Level for a User](#). You cannot 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 repository, then the logging level for the administrator is 0.

Set the logging level based on the amount of logging that is 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	<p>Logs the SQL statement issued from the client application. Also logs the following:</p> <ul style="list-style-type: none"> 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, execute, and fetch the last record. This matches the <code>TOTAL_TIME_SEC</code> that is logged in usage tracking, as described in Description of the Usage Tracking Data. 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	<p>Logs everything logged in Level 1.</p> <p>Additionally, for each query, logs the repository 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.</p>
Level 3	<p>Logs everything logged in Level 2.</p> <p>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.</p> <p>Do not select this level without the assistance of Oracle Support Services.</p>

Logging Level	Information That Is Logged
Level 4	Logs everything logged in Level 3. Additionally, logs the query execution plan. Do not 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. Do not select this level without the assistance of Oracle Support Services.
Level 6 and 7	Not used.

Setting the Query Logging Level for a User

You can configure the amount of query data to log per user.

1. In the Oracle BI 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**.

Using the Log Viewer

Use the Oracle Business Intelligence 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.

Running 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 UNIX 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 Oracle Business Intelligence repository. 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 does not exist, then a new file is created. If this argument is not 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. See *Security Guide for Oracle Business Intelligence Enterprise Edition* for information.

Administrators can view the query log using the **Manage Sessions** option in the Presentation Services Administration page.

Interpreting 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 repository, 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.

Logging in Oracle BI Server

This section describes logging specifically in Presentation Services .

Topics include:

- [Using the Oracle BI Presentation Services Logging Facility](#)
- [Setting the Logging Levels for Oracle BI Presentation Services](#)
- [Structure for the Oracle BI Presentation Services Configuration File](#)
- [Examples of the Formats of Logged Messages](#)
- [Oracle BI Presentation Services Message Structure](#)
- [Oracle BI Presentation Services Log Filters](#)
- [Diagnosing Issues with Agents](#)

See [Understanding Diagnostic Log Files and Log Configuration Files](#).

Using the Oracle BI Presentation Services Logging Facility

You can troubleshoot issues and errors using the Oracle BI Presentation Services logs.

By default, Oracle BI 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 sawlogxx.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 Oracle BI 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 Oracle BI Presentation Services configuration information is loaded from the instanceconfig.xml file.

Caution:

Because logging affects performance, do not increase the logging on a production implementation, except to diagnose specific issues.

Setting the Logging Levels for Oracle BI Presentation Services

You use options on the Administration page in Presentation Services to affect logging levels.

1. In the global header, click **Administration**.
2. In the Maintenance and Troubleshooting area, select the logging level to use under Reload Log Configuration.
3. Click **Reload Log Configuration** to allow the change to take effect without restarting Presentation Services.

The change remains in effect even when you stop and restart Presentation Services.

4. Click the **Manage Sessions** link to display the Manage Sessions page.
5. For each session, specify the appropriate level in the **Log Level** column of the table.

The updated level takes effect immediately for that session. When you select a level, ensure that its severity value is smaller than or equal to the value specified for all messages in Presentation Services.

Structure for the Oracle BI Presentation Services Configuration File

The structure of the Oracle BI 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 6-3 Structure of Log 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 6-4 instanceconfig.xml File with Four Writers

```
<?xml version="1.0" ?>
<Server>
. . . . .
<Logging>
<Writers>
<Writer implementation="FileLogWriter" name="Global File Logger"
writerClassId="1" dir="{%ORACLE_BIPS_INSTANCE_LOGDIR%}" filePrefix="sawlog"
maxFileSizeKb="10000" filesN="10" fmtName="ODL-Text"
ODLLogFilePath="{%ORACLE_BIPS_INSTANCE_LOGDIR%}/diagnostic.log"/>
<Writer implementation="CoutWriter" name="Global Output Logger"
writerClassId="2" />
<Writer implementation="EventLogWriter" name="Event Logger"
writerClassId="3" />
<Writer implementation="CrashWriter" name="CrashWriter"
```

```

writerClassId="4"
/>
</Writers>
<WriterClassGroups>
<WriterClassGroup name="All">1,2,3,4</WriterClassGroup>
<WriterClassGroup name="File">1</WriterClassGroup>
<WriterClassGroup name="Console">2</WriterClassGroup>
<WriterClassGroup name="EventLog">3</WriterClassGroup>
<WriterClassGroup name="Crash">4</WriterClassGroup>
</WriterClassGroups>
<Filters>
<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" />
<FilterRecord writerClassGroup="File" path="saw.mktgsqsubsystem.joblog"
information="1" warning="2" error="31" trace="0" incident_error="32"/>
<FilterRecord writerClassGroup="File" path="saw.httpservlet.request"
information="16" warning="32" error="32" trace="0" incident_error="32"/>
<FilterRecord writerClassGroup="File" path="saw.httpservlet.response"
information="16" warning="32" error="32" trace="0" incident_error="32"/>
</Filters>
</Logging>
</Server>

```

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	disableCentralControl	(Optional) Determines that this entry is not updated by Fusion Middleware Control. Default value is true.

Element	Attribute	Description
Writer	implementation	<p>The following implementations are defined:</p> <ul style="list-style-type: none"> • FileLogWriter. Writes to a disk file. • CoutWriter. Writes to standard output. • EventLogWriter. Writes to a Windows event log or UNIX syslog. • CrashWriter. A Windows only facility that writes to a crash dump file when Presentation Services attempts to log from a specific source file and line number. <p>Used in a production environment for information of some loggable but irrecoverable error (for example, failed NQTEST).</p> <p>Note: Use this implementation with care as it might leave the server in an unstable state. Use this implementation in very rare diagnostic-only scenarios on a test system. On Windows, CrashWriter requires the appropriate version of <code>dbghelp.dll</code> (at least 6.0.17.0).</p> <p>The correct <code>dbghelp.dll</code> can be found in <code>support/windows/system64</code>.</p> <p>Put this DLL in the <code>WINNT/system64</code> or in the <code>main/bin</code> directory.</p> <p>No registration is required.</p>
Writer	name	Unique name for the writer.
Writer	writerClassId	<p>Specifies an integer number in the range 1 through 10. This number is used by filters to allow or prohibit logging.</p> <p>Each distinct writer must have a unique value, which is used later for filter configuration.</p> <p>Different writers might have the same class ID, but if they do, those writers cannot be distinguished by filters.</p>
Writer	fmtName	<p>(Optional) Specifies the format of logged messages. Valid values are:</p> <ul style="list-style-type: none"> • default - 10g style. Formats messages with identifying headings. • ODL-TEXT. Formats messages in Oracle Diagnostic Text format. • ODL-XML. Formats messages in Oracle Diagnostic XML format. <p>If you do not set this attribute, then logged messages are displayed in the default format which for file log writers is 10g style and for console is ODL-TEXT.</p> <p>See Examples of the Formats of Logged Messages for examples.</p>
Writer (FileLogWriter specific attribute)	dir	Specifies the directory where log files are created.

Element	Attribute	Description
Writer (FileLogWriter specific attribute)	ODLLogFilePath	Specifies the file that Fusion Middleware Control displays in the Log Viewer.
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.
Writer (EventLogWriter specific attribute)	winSource	Specifies the event log source for logged events.
Writer (CrashWriter specific attribute)	file	Specifies the dump file path. On Windows, a dump file is created in <code>bin/coredumps</code> and Presentation Services continues to run.
Writer (CrashWriter specific attribute)	line	Dump file line number.
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	disableCentralControl	(Optional) Determines that this entry is not updated by Fusion Middleware Control. Default value is true.
FilterRecord	path	Specifies the log source path. To enable the logging of SOAP information, enter the following value: <code>saw.httpservlet.request.soaprequest</code> The current filter record is applied to the software component that is 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 (10g style), ODL-TEXT, and ODL-XML.

The following example shows the default format.

Example 6-5 Default Format

The default format generates messages with identifying headings, such as:

```
Type: Information
Severity: 30
Time: Wed Jul 26 11:22:20 2006
File: project\sawserver\sawserver.cpp
Line: 399
Properties: ThreadID-2552
Location:
    saw.sawserver
    saw.sawserver.initializesawserver
    saw.sawserver
Oracle BI Presentation Services has started successfully.
```

The following example shows the ODL-TEXT format.

Example 6-6 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
]]

[2010-05-27T10:51:20.000-07:00] [OBIPS] [NOTIFICATION:1] [] [saw.sawserver] [ecid:
1243446680218334471555761] [tid: 2552] Oracle BI Presentation Services (OBIPS)
11.1.1.2 (Build 0) are starting up.[]
```

```
File:sawserver.cpp
Line:432
Location:
    saw.sawserver
    saw.sawserver.initializesawserver
    saw.sawserver
ecid: 1243446680218334471555761
]]
```

The following example shows the ODL-XML format.

Example 6-7 ODL-XML Format

The xml format generates messages in XML format, such as:

```
<msg time="2010-05-08T18:41:05.000+00:00"
comp_id="OBIPS" type="NOTIFICATION" level="1" msg_id=""
module="saw.sawserver" ecid="124180446517874242628761" tid="127c">
<txt> Oracle BI Presentation Services has started successfully</txt>
<suppl_detail />
</msg>
```

Oracle BI Presentation Services Message Structure

Each message that is logged by BI Publisher 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 is not important at all.
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.

Oracle BI 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" />
<FilterRecord writerClassGroup="File" path="saw.mktgsqlsubsystem.joblog"
information="1" warning="2" 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" />
<FilterRecord writerClassGroup="File" path="saw.mktgsqlsubsystem.joblog"
information="1" warning="2" error="31" trace="0" incident_error="32"/>
```

Diagnosing Issues with Agents

This section contains the following topics:

- [Debugging Agents Using Fusion MiddlewareControl](#)
- [Manually Debugging Agents Using instanceconfig.xml](#)

Debugging Agents Using Fusion MiddlewareControl

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 [Using 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	TRACE: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

Event	State	Message Type:Level
Agent Chain Complete	Cancelled	NOTIFICATION:1
Agent Chain Complete	Try Again	NOTIFICATION:1
Agent Chain Complete	Completed	TRACE: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 [Configuring Log File Rotation Policy and Specifying Log Levels](#). You can also filter log entries using ECID to find information specific to a particular agent chain.

Manually Debugging Agents Using `instanceconfig.xml`

If an agent fails to execute 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 `Debug` element to `true` in the Oracle BI Scheduler `instanceconfig.xml` file. For example, located at:

```
oraclehome/user_projects/domains/bi/config/fmwconfig/biconfig/
OBISCH
```

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 Oracle Business Intelligence 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 execute. You can use a text editor to view the files. Entries are generally self-explanatory.

The presence of an error log does not 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.

You can also view error messages and exit codes for job instances in Job Manager. See Instance Properties in Job Manager in *Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*. Exit status shows the number of deliveries successfully completed.

Using ODBC/JDBC Procedures to Obtain Oracle BI Server Diagnostics

This section describes how to use ODBC/JDBC procedures to obtain diagnostic information for the Oracle BI Server.

This section contains the following topics:

- [About the Oracle BI Server ODBC/JDBC Procedures](#)
- [Obtaining a List of Available Diagnostic Categories](#)
- [Running Specific Diagnostics](#)
- [About Parameters for ODBC/JDBC Procedures](#)

About the Oracle BI Server ODBC/JDBC Procedures

You can use ODBC/JDBC procedures to obtain diagnostic information for the Oracle BI Server.

These procedures are especially useful on non-Windows platforms where you cannot run the Administration Tool.

Use the `nqcmd` utility to run the procedures using ODBC. See *Using nqcmd to Test and Refine the Repository in Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

You can also run the procedures using JDBC. For more information about using JDBC to connect to the Oracle BI Server, see the `README.TXT` file contained in the `bijdbc.jar` file in `ORACLE_HOME/bi/bifoundation/jdbc`.

Obtaining 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.
DBInstance:DBNAME1	All of the statistics related to the DB instance named in DBNAME1
DBInstance:DBNAME _n	All of the statistics related to the DB instance named in DBNAME _n

Category	Description
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.

Running 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 execute 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.

Parameter Name	Description
ROWS PER SEC	The number of rows retrieved each second from the back-end database.
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.

Parameter Name	Description
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.
AVG MISS	The average number of misses each second for the specified cache object.

Troubleshooting System Startup

Startup errors require specific troubleshooting procedures.

This section contains solutions that are related to system startup:

- [Administration Server Fails to Start When the Database Is Not Running](#)
- [Managed Server Is Down](#)
- [Oracle BI Server Fails to Start](#)
- [Cannot Log In](#)

Administration Server Fails to Start When the Database Is Not Running

When you start the Administration Server, the repository database that was specified during installation must be running, or else you see JDBC errors that prevent startup.

Problem: The Administration Server fails to start.

If the Administration Server fails to start, then:

- View the Administration Server and Managed Server log files in the following directory:

```
BI_DOMAIN/servers/AdminServer/logs
```

- You can also check the Managed Server log files in the following directory:

```
BI_DOMAIN/servers/bi_server1/logs
```

Cause: Database Down: in AdminServer.log, "Caused By: java.net.UnknownHostException: yourcomputername" deep in the trace from:

```
####<Jan 19, 2010 8:04:09 PM PST> <Info> <JDBC> <username> <AdminServer>
<[ACTIVE] ExecuteThread: '0' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <Stack trace associated with message 001129 follows:
```

```
java.sql.SQLException: The Network Adapter could not establish the connection.
```

Resolution: Start the database.

Managed Server Is Down

If the Managed Server is down, then use the command line to start it.

See [Starting Oracle Business Intelligence Component Processes in a Domain](#).

Oracle BI Server Fails to Start

If the BI Server fails to start, then view the log files.

View log files in the Log Viewer or in the following directory:

BI_DOMAIN/servers/obis1/logs, or use the Log Viewer.

Cannot Log In

Problem: Cannot log in to Oracle WebLogic Server Administration Console.

Cause: The Administration Server is not running.

Resolution: Check to see if `http://<host>:<port>/console` starts. If not, start the BI system. See [Using Commands to Start, Stop, and View Status of Oracle BI EE Processes](#).

7

Managing Usage Tracking

This chapter describes how to manage usage tracking for Oracle Business Intelligence. The Oracle BI Server supports the collection of usage tracking data. When usage tracking is enabled, the Oracle BI Server collects usage tracking data for each query and inserts it directly to a database table.

Note:

The Oracle BI Summary Advisor feature works in conjunction with the usage tracking feature. Summary Advisor only works with direct insertion usage tracking.

Oracle BI Summary Advisor is only available when you are running Oracle Business Intelligence on the Oracle Exalytics Machine. See Using Oracle BI Summary Advisor to Identify Query Candidates for Aggregation in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

This chapter includes the following sections:

- [About Usage Tracking](#)
- [Setting Up Direct Insertion to Collect Information for Usage Tracking](#)
- [Description of the Usage Tracking Data](#)

About Usage Tracking

The Oracle BI Server supports the accumulation of 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.

The BI Server tracks usage at the detailed query level.

When you enable usage tracking, statistics for every query are inserted into a database table or are written to a usage tracking log file. If you use direct insertion, then the BI Server directly inserts the usage tracking data into a relational database table. It is recommended that you use direct insertion to write statistics to a database table.

When the BI Server starts, it validates the column names in the metadata against the list of valid columns in the usage tracking table. The following events occur:

- **Column names.** If there is a mismatch between the columns in the database table and the columns in the metadata, then it results in a database error on insert.
- **Varchar length.** If the length in the metadata and the set length in the table do not match, then an error is written to the nqserver.log file and usage tracking is disabled.

Setting 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 set up direct insertion, and contains the following topics:

- [Setting Up the Usage Tracking Statistics Database](#)
- [Setting Direct Insertion Parameters](#)
- [Setting Optional Direct Insert Parameters](#)

Setting 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 Business Intelligence 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 [Description of the Usage Tracking Data](#).

You also need to import the database into the Physical layer of the Oracle BI repository.

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 Business Intelligence for usage tracking statistics, because this database has the RCU-created tables already.
2. Open the Administration Tool and import the database into the Physical layer. See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
3. Save and close the repository.

Setting 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 Oracle 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 Oracle BI repository. 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 Oracle BI repository. 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 Oracle BI repository. 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 Oracle BI repository. 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 is recommended that the connectivity type supports international data.

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

Setting 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 do not 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 [NQSSConfig.INI File Configuration Settings](#).

Description of the Usage Tracking Data

The table below describes each column in the `S_NQ_ACCT` usage tracking table. Where appropriate, the data type and length is also included.

As you review the descriptions in the table below, you might assume that certain of the 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 following list explains why the columns do not provide such exact values:

- The various processes run in parallel and their speed depends on the load on the BI Server and on database performance. The 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 configuration of the BI Server.

Column	Description
CACHE_IND_FLG	Default is N. Y indicates a cache hit for the query; N indicates a cache miss.
COMPILE_TIME_SEC	The time in seconds that is required to compile the query. The number for <code>COMPILE_TIME_SEC</code> is included in <code>TOTAL_TIME_SEC</code> , as described in this table.
CUM_DB_TIME_SEC	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, if 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. Since the queries run in parallel, <code>ngsserver</code> is only connected to the database for 15 seconds, but <code>CUM_DB_TIME_SEC</code> will show 45 seconds.
CUM_NUM_DB_ROW	The total number of rows that are returned by the back-end databases.
END_DT	The date the logical query was completed.
END_HOUR_MIN	The hour and minute the logical query was completed.

Column	Description
END_TS	<p>The date and time that the logical query finished. The start and end timestamps also reflect any time that the query spent waiting for resources to become available.</p> <p>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, and this is recorded as the end_ts time.</p>
ERROR_TEXT	<p>Default is Null. Varchar(250)</p> <p>Error message from the back-end database. This column is only applicable if the SUCCESS_FLG (for more information, see entry later in this table) is set to a value other than 0 (zero). Multiple messages are concatenated and are not parsed by the BI Server.</p>
ID	The unique row ID.
NODE_ID	<p>Concatenates <hostname>:<component_name> where <component_name> can be overridden by the environment variable COMPONENT_NAME. For example, examplehost:obis1 (for a single instance).</p> <p>Default value of COMPONENT_NAME is obis1.</p>
NUM_CACHE_HITS	<p>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).</p> <p>Default is Null.</p>
NUM_CACHE_INSERTED	<p>Indicates the number of times that the query generated a cache entry.</p> <p>Default is Null. NUM_CACHE_INSERTED is a 32-bit integer (or a 10-digit integer).</p>
NUM_DB_QUERY	The number of queries that were submitted to back-end databases to satisfy the logical query request. For successful queries (SuccessFlag = 0) this number is 1 or greater.
PRESENTATION_NAME	<p>Default is Null. Varchar(128)</p> <p>The name of the Oracle BI Presentation Catalog.</p>
QUERY_BLOB	<p>Contains the entire logical SQL statement without any truncation.</p> <p>The QUERY_BLOB column is a long character string.</p>
QUERY_KEY	<p>Default is Null. Varchar(128).</p> <p>An MD5 hash key that is generated by Oracle Business Intelligence from the logical SQL statement.</p>
QUERY_SRC_CD	<p>The source of the request.</p> <p>Values that can be inserted, for example:</p> <p>An analysis, or any export operation inserts 'Report'.</p> <p>Using the 'Value' drop down in a filter dialog, or using a dashboard prompt inserts 'ValuePrompt'.</p> <p>Agent to seed analytics server cache inserts 'Seed'.</p> <p>Online Admin Tool physical table or column row count, or view data inserts 'NULL'.</p>

Column	Description
QUERY_TEXT	Varchar(1024). The SQL statement that was submitted for the query. You can change the length of this column (using the ALTER TABLE command), but note that the text that is written into this column is always truncated to the size that is defined in the physical layer. It is the responsibility of the repository administrator not to set the length of this column to a value greater than the maximum query length that is 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. Hence, if you use a multi-byte character set, the actual maximum string size has a varying number of characters, depending on the character set and characters used.
REPOSITORY_NAME	The name of the repository that the query accesses.
ROW_COUNT	The number of rows that are returned to the query client. When a large amount of data is returned from a query, this column is not populated until the user displays all of the data.
IMPERSONATOR_USER_NAME	Default is None. Varchar(128). The user name of the impersonated user. If the request is not run as an impersonated user, then the value is 'None'.
SAW_DASHBOARD	The path name of the dashboard. If the query was not submitted through a dashboard, then the value is NULL.
SAW_DASHBOARD_PG	Default is Null. Varchar(150) The page name in the dashboard. If the request is not a dashboard request, then the value is NULL.
SAW_SRC_PATH	The path name in the Oracle BI Presentation Catalog for the analysis.
START_DT	The date that the logical query was submitted.
START_HOUR_MIN	The hour and minute that the logical query was submitted.
START_TS	The date and time that the logical query was submitted.
SUBJECT_AREA_NAME	The name of the business model that is being accessed.
SUCCESS_FLG	The completion status of the query, as defined in the following list: <ul style="list-style-type: none"> • 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.
TOTAL_TIME_SEC	The time in seconds that the BI Server spent working on the query while the client waited for responses to its analyses. TOTAL_TIME_SEC includes the time for COMPILE_TIME_SEC. This setting is the same as the Response time in the nquery.log file, as described in Setting the Query Logging Level .
USER_NAME	The name of the user who submitted the query.

The table below describes the S_NQ_DB_ACCT table, which supplements the usage tracking table by providing the physical SQL information for the logical queries stored in S_NQ_ACCT. S_NQ_DB_ACCT has a foreign key relationship back to S_NQ_ACCT.

Column	Description
END_DT	The date the physical query was completed.
END_HOUR_MIN	The hour and minute the physical query was completed.
END_TS	The date and time the physical query finished. The start and end timestamps also reflect any time that the query spent waiting for resources to become available.
ID	The unique row ID.
LOGICAL_QUERY_ID	Varchar2(50). Refers to the logical query in the S_NQ_ACCT table.
QUERY_BLOB	Contains the entire physical SQL statement without any truncation. The QUERY_BLOB column is a long character string.
QUERY_TEXT	Varchar(1024). The SQL statement that was submitted for the query.
ROW_COUNT	The number of rows that are returned to the query client.
TIME_SEC	The physical query execution time.
START_DT	The date that the physical query was submitted.
START_HOUR_MIN	The hour and minute that the physical query was submitted.
START_TS	The date and time that the physical query was submitted.

The table below describes each column in the S_NQ_INITBLOCK usage tracking table, which tracks information about initialization blocks.

Column	Description
USER_NAME	Varchar2(128). The name of the user who ran the initialization block.
REPOSITORY_NAME	Varchar2(128). The name of the repository that the query accesses.
TENANT_ID	Varchar2(128). The name of the tenant of the user who ran the initialization block.
SERVICE_NAME	Varchar2(128). The name of the service.
ECID	Varchar2(1024). The system-generated execution context ID.
SESSION_ID	Number(10). The ID of the session.
BLOCK_NAME	Varchar2(128). The name of the initialization block that was executed.
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 any time that the query spent waiting for resources to become available.
DURATION	Number(13,3). The length of time it took to execute the initialization block.

Column	Description
NOTES	Varchar2(1024). Notes about the initialization block and its execution.

Part V

Configuring Oracle Business Intelligence

Although the installer installs Oracle Business Intelligence and the configuration assistant provides a functional sample application, 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 Business Intelligence to your environment and user needs.

This part includes the following chapters:

- [Using Tools for Managing and Configuring Oracle Business Intelligence](#)
- [Managing Metadata and Working with Service Instances](#)
- [Configuring Connections to External Systems](#)
- [Configuring Presentation Setting Defaults](#)
- [Configuring Mapping and Spatial Information](#)
- [Configuring Time Zones](#)
- [Localizing Oracle Business Intelligence](#)
- [Configuring Currency Options](#)
- [Configuring and Managing the Oracle BI Presentation Catalog](#)

8

Using Tools for Managing and Configuring Oracle Business Intelligence

This chapter introduces the tools to manage and configure Oracle Business Intelligence including Fusion Middleware Control, WebLogic Server Administration Console, a text editor, and the WebLogic Scripting Tool.

This chapter includes the following sections:

- [Why Use Fusion Middleware Control and WebLogic Server Administration Console?](#)
- [Managing Oracle Business Intelligence System Components Using Fusion Middleware Control](#)
- [Configuring Oracle Business Intelligence System Settings](#)
- [Managing Oracle Business Intelligence Java Components Using the Oracle WebLogic Server Administration 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 Oracle Business Intelligence system.

These Web-based tools support the most common system administration tasks for Oracle Business Intelligence. See [Getting Started with Managing Oracle Business Intelligence](#).

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 Oracle Business Intelligence System Administration](#).

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 concurrent updates of the same configuration settings by multiple administrators 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 is therefore recommended that multiple administrator users do not share the same administrator account.

Managing Oracle Business Intelligence System Components Using Fusion Middleware Control

You can use Fusion Middleware Control to manage, monitor, and configure Oracle Business Intelligence system components (for example, the Oracle BI Server, Oracle BI 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:

- [Logging into Fusion Middleware Control](#)
- [Displaying Oracle Business Intelligence Pages in Fusion Middleware Control](#)
- [Using the Navigation Tree in Fusion Middleware Control](#)
- [Tips for Using Fusion Middleware Control with Oracle Business Intelligence](#)

Logging 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 for Starting the Oracle Business Intelligence 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 Business Intelligence.

Alternatively, enter any other user name and password that has been granted the Oracle BI Administrator application role. Fusion Middleware Control is displayed.

 **Note:**

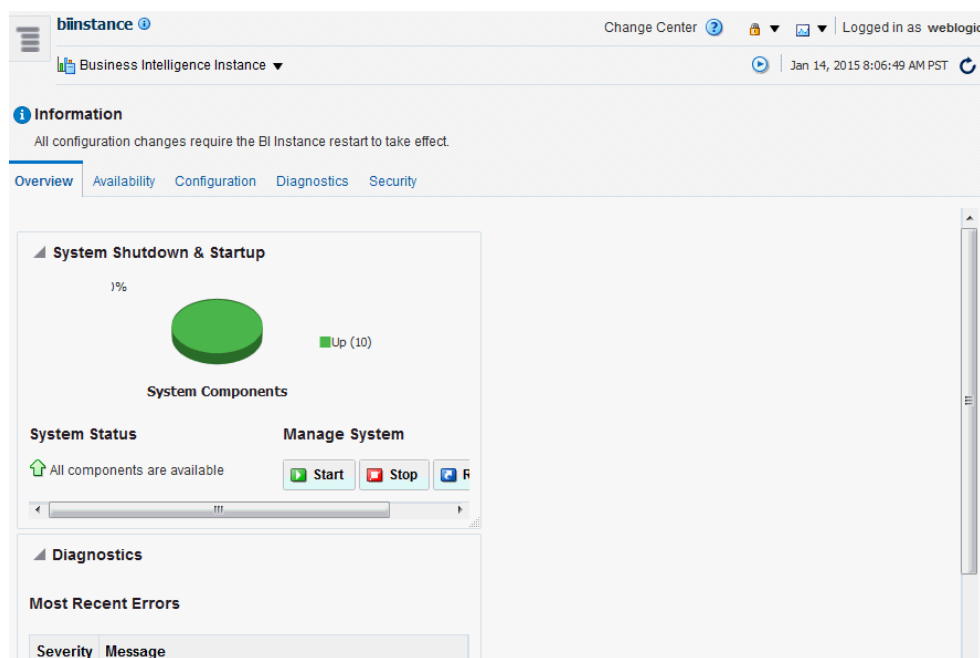
If you have the browser configured to send HTTP requests to a proxy server, then you might have to configure the browser to not send Administration Server HTTP requests to the proxy server. If the Administration Server is on the same computer as the browser, then ensure that requests that are sent to localhost or 127.0.0.1 are not sent to the proxy server.

Displaying Oracle Business Intelligence Pages in Fusion Middleware Control

Use this topic to display Oracle Business Intelligence pages that enable you to manage Oracle Business Intelligence 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.



 **Note:**

If the Business Intelligence folder is not visible or there is no **biinstance** node under it, then Oracle Business Intelligence system components have not been installed. See *Installing and Configuring Oracle Business Intelligence*.

The Overview page displays the current system status, by providing information about availability, and issues identified within the BI domain (see [What Is the Oracle Business Intelligence System Logical Architecture?](#)). The Overview page also enables you to start and stop Oracle Business Intelligence.

3. From the Overview page, select an appropriate tab to perform Oracle Business Intelligence management tasks.

Using 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 Business Intelligence for Microsoft Office, Oracle BI 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 Business Intelligence 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 Business Intelligence system components that can be managed using Fusion Middleware Control.

Select this node to display the Overview page and manage the system components.

- **Metadata Repositories**

This node represents the Metadata Services (MDS) schema repositories that can be managed using Fusion Middleware Control.

Tips for Using Fusion Middleware Control with Oracle Business Intelligence

There are several considerations to keep in mind when using Fusion Middleware Control with Oracle Business Intelligence.

Keep the following tips in mind as you use Fusion Middleware Control to manage Oracle Business Intelligence:

- 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 Business Intelligence within Fusion Middleware Control but not make any changes. You can configure such a user by making him a member of the Monitors group. See *Securing Resources Using Roles and Policies for Oracle WebLogic Server*.
- You might encounter display problems when using Internet Explorer 8 with Fusion Middleware Control. For example, scroll bars might be missing on the Log Messages tab of the Diagnostics page, even when the bars are required to see all the text.

To work around this issue, ensure that Compatibility View mode is turned off for the browser.

1. From the Tools menu, select **Internet Options**. On the Advanced tab in the Browsing section, ensure that **Automatically recover from page layout errors with Compatibility View** is not checked.
2. From the Tools menu, select **Compatibility View Settings**. Ensure that **Display intranet sites in Compatibility View** and **Display all websites in Compatibility View** are not checked.

Configuring Oracle Business Intelligence System Settings

You configure the Oracle Business Intelligence system settings by changing values stored in domain-specific locations related to either functional behavior (for example, cache, thresholds), or environmental settings (for example, host names, ports, files or metadata locations).

You can use the following methods:

- [Using Fusion Middleware Control](#)
- [Using a Text Editor](#)
- [Using the WebLogic Scripting Tool \(WLST\)](#)

The table below shows which method to use when configuring Oracle Business Intelligence 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.	Fusion Middleware Control See Using Fusion Middleware Control . Oracle recommends that you use this method. However, if a setting is not available, you can use a text editor.	Change values in specific Oracle Business Intelligence configuration pages in Fusion Middleware Control. For example, to enable the BI Server cache, you click the Cache Enabled check box (select/clear) in the Performance tab of the Configuration page.

What Do You Want to Do?	What Methods Can You Use?	How Are Updates Made?
Change configuration settings by manually editing a file.	Text editor See Using a Text Editor . Oracle recommends that you use this method when a setting is not available in Fusion Middleware Control Oracle Business Intelligence pages.	Change values in a configuration text file using a text editor.
Make more complex configuration changes using a scripting tool.	WebLogic Scripting Tool (WLST) See Using the WebLogic Scripting Tool (WLST) . Oracle recommends that you use this method when instructed by the documentation.	Make configuration changes by running commands using the WLST scripting tool.

Using Fusion Middleware Control

You can use Fusion Middleware Control to update specific Oracle Business Intelligence configuration settings.

Configuration settings you can change include performance settings, dashboard and analysis default presentation settings, and mail server settings used by agents.

If an Oracle Business Intelligence configuration setting is not available in Fusion Middleware Control, you can use a text editor to update the setting in a configuration file. See [Configuring Oracle Business Intelligence System Settings](#).

1. Go to the Business Intelligence Overview page, as described in [Displaying Oracle Business Intelligence 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 do not 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 execute 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**.

Using 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 Oracle Business Intelligence in Fusion Middleware Control, go to the Process tab in the Availability page, and restart all components.

See [About Managing Oracle Business Intelligence Processes](#).

Using the WebLogic Scripting Tool (WLST)

Oracle provides WebLogic Scripting Tool (WLST) scripts to perform Oracle Business Intelligence WebLogic configuration tasks. For example, to create a domain during installation, or to add a machine for high availability.

You run WLST scripts commands from the following location:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh (wlst.cmd on Windows)
```

You can configure Oracle Business Intelligence system settings in online (system processes running) or offline (system processes stopped) mode. The following sections describe the different conditions that apply when making online and offline configuration changes.

- [Making Offline Configuration Changes](#)
- [Making Online Configuration Changes](#)

Making Offline Configuration Changes

All offline configuration changes must be made on the master host, except those relating to node managers, and all Oracle BI EE processes must be stopped first.

To consume offline changes you must start the Administration server, Managed servers, and then system components (in that order). This is the general requirement for any configuration change as the command to start the managed server is the only process that replicates the configuration, from a running Administration Server.

Assumptions and pre-requisites:

- You must have file system (offline) or Weblogic Administrator (online) permissions
- When you start component(s) offline, the Administration server and node managers must be started first, see [Starting Oracle Business Intelligence Component Processes in a Domain](#)

Running WLST Offline

You run WLST scripts commands from the following location:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh
```

- Before issuing offline commands you must select a domain using the `readDomain(DOMAIN_HOME)` command.

For example:

```
readDomain('/u01/bi')
```

- After you have issued your offline commands you must commit the changes using the `updateDomain()` command.

For example:

```
updateDomain('/u01/bi')
```

- De-select the domain using the `closeDomain()` command.

For example:

```
closeDomain('/u01/bi')
```

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

Making Online Configuration Changes

You can make online configuration changes from any computer where you have access to the Administration Server domain mbeans.

Assumptions and pre-requisites:

- You must have Weblogic Administrator permissions.
- The Administration Server must be running.
- After making changes, you must restart the affected BI component(s), see [Starting Oracle Business Intelligence Component Processes in a Domain](#).

Running WLST Online

You run WLST script commands from the following location:

```
ORACLE_HOME/oracle_common/common/bin/wlst.sh
```

- Connect to Administration Server by issuing the following command:

```
./wlst.sh connect(<username>, <password>, <connect string>)
```

For example,

```
./wlst.sh connect('weblogic', 'mypassword', 't3://localhost:9500')
```

- You must enter edit tree and start an edit session by issuing edit() and startEdit() commands before issuing commands.
- Use the save() command to save all changes made during the edit session.
- Use the activate() command to commit your changes.

If you attempt to issue a command outside an edit session, the command will fail and display a help message.

If you make a mistake or decide to abandon changes then you must use the undo() or cancelEdit() command.

Updating the Java Development Kit (JDK)

After you install and configure Oracle Business Intelligence, 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, as described in the system requirements and certification documentation. See [System Requirements and Certification](#).

1. Stop all services for Oracle Business Intelligence.
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 `jdkversion-num` directory.
5. Rename the directory from `jdkversion-num` to jdk, to ensure that all existing configuration references remain valid.
6. Restart the services for Oracle Business Intelligence.

For information on installing with a specific JDK, see *Installing and Configuring Oracle Business Intelligence*.

Managing Oracle Business Intelligence Java Components Using the Oracle WebLogic Server Administration Console

You use the Oracle WebLogic Server Administration Console to manage Oracle Business Intelligence 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 Oracle Business Intelligence Processes](#).

1. If the Administration Server for WebLogic Server is not 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 have 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 Business Intelligence. 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 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. If the Administration Server is on the same computer as the browser, then ensure that requests sent to localhost or 127.0.0.1 are not sent to the proxy.

In Oracle WebLogic Server Administration Console you select the bi domain, as shown in the figure below.

The screenshot displays the Oracle WebLogic Server Administration Console interface. The top navigation bar includes 'Home', 'Log Out', 'Preferences', 'Record', and 'Help'. The main content area is titled 'Settings for bi' and features several tabs: 'Configuration', 'Monitoring', 'Control', 'Security', 'Web Service Security', 'Patching', and 'Notes'. Under the 'Configuration' tab, there are sub-tabs for 'General', 'JTA', 'Concurrency', 'JPA', 'EJBs', 'Web Applications', 'Logging', 'Log Filters', and 'Batch'. The 'General' sub-tab is active, showing a 'Name' field with the value 'bi' and a checked 'Enable Administration Port' checkbox. The 'Administration Port' field is set to '9002'. A 'Save' button is located at the top left of the configuration area. The left sidebar contains a 'Change Center' with 'View changes and restarts' and 'Release Configuration' buttons, a 'Domain Structure' tree with 'bi' selected, and a 'How do I...' section with links to 'Change Console preferences' and 'Configure the domain-wide administration port'.

You can monitor and manage Oracle Business Intelligence 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 Business Intelligence is not supported, because many context roots are used for internal links and end-user end points.

9

Managing Metadata and Working with Service Instances

This chapter describes how to manage Oracle Business Intelligence metadata in BI Application Archive (BAR) files, and how to work with Service Instances. This chapter includes the following sections:

- [About Oracle Business Intelligence Application Archive \(BAR\) Files](#)
- [Managing Service Instances](#)

About Oracle Business Intelligence Application Archive (BAR) Files

This topic contains the following sections:

- [What Are Oracle Business Intelligence Application Archive \(BAR\) Files?](#)
- [What Predefined BAR Files are Available?](#)
- [About Importing BAR Files](#)

What Are Oracle Business Intelligence Application Archive (BAR) Files?

An Oracle Business Intelligence application archive (BAR) file is a compressed archive file that contains a cohesive set of BI metadata artifacts (data model, content model, and authorization model).

A BAR file can be imported into a BI Service Instance. You can backup a BI Service Instance into a BAR file, and subsequently restore it either to the existing service instance running in the BI domain, or into a different service instance running on a different BI installation.

A BAR file contains the following BI application module artifacts:

- Data model metadata for the Oracle BI Server. This metadata is xml-based but functionally equivalent to a .RPD file.
- Presentation Services catalog metadata for a service instance.
- Security policy metadata containing application role and application role memberships plus permission and permission set grants for a service instance.
- A manifest file declaring the dependencies for the BAR file.

What Predefined BAR Files are Available?

Oracle provides a number of predefined BAR files, containing BI metadata to use with a service instance.

You can choose one of the following BAR files during installation:

- Empty BAR - oracle.bi.application.empty.bar

The empty BAR file enables you to have complete control over metadata and security policy, or you may simply have an existing BAR from another system that you might want to import.

```
ORACLE_HOME/bi/bifoundation/admin/provisioning/
oracle.bi.application.empty.bar
```

The Empty BAR file contains only enough permissions for the administrator user to login and administer the system.

The following table shows the application role and corresponding permission sets that come with the Empty BAR file.

Application Role	Permission Set
BIServiceAdministrator	obisch.administrator essbase.administrator obips.administrator cds.administrator bip.administrator obis.administrator

- SampleAppLite BAR - SampleAppLite.bar

You can select SampleAppLite at install time (see *Installing and Configuring Oracle Business Intelligence*).

The SampleAppLite BAR enables you to start using BI Analytics straight away, enabling you to demonstrate the functionality of the product.

The SampleAppLite BAR file contains a file based data source with 2 subject areas, a collection of reports, and application roles for read, write, and administration tasks.

```
ORACLE_HOME/bi/bifoundation/samples/sampleapplite/
SampleAppLite.bar
```

The following table shows the application roles and corresponding permission sets that come with the SampleAppLite BAR file.

Application Role	Permission Set
BIServiceAdministrator	obisch.administrator essbase.administrator obips.administrator cds.administrator bip.administrator obis.administrator
BIConsumer	bip.consumer
BIContentAuthor	va.author obisch.author bip.author

See [Working with the Sample Application](#).

- Starter BAR

You use the starter BAR when you want to build metadata from the beginning, but need a typical security policy to be in place.

The starter BAR is essentially empty except for containing the same security policy as the Sample application BAR.

For example, if you want to create a service instance for Visual Analyzer where users will import their own data from scratch, by importing the starter BAR into your service instance it means that you only have to provision users with the starter application roles and they can immediately login and start using the product.

About Importing BAR Files

When you import a BAR file into a service instance, the service instance will use the data model, content and security policy imported from the BAR file.

The import process does the following:

- Takes the content, model and security policy defined by the BAR, and deploys it to a Service Instance.
- Overwrites any existing content, model, security policy already established in the Service Instance.

You can import a BAR file exported from other 12c releases as described in [importServiceInstance Command](#). BAR files exported from 12.2.1.3.0 can be imported into another 12.2.1.3.0 environment, or into an Oracle Analytics Cloud environment.

BAR files exported from 11g shouldn't be imported into 12c — instead use the migration process as described in *Upgrading Oracle Business Intelligence*.

Managing Service Instances

A service instance contains all of the Oracle Business Intelligence metadata (that is, repository data, presentation catalog, security policy), and includes the customizations that you make to the metadata. You manage a service instance in an BI domain using WebLogic Scripting Tool (WLST) commands described in the table below.

See [Using the WebLogic Scripting Tool \(WLST\)](#).

Command	Description
listBIServiceInstances	This command lists all Service Instance keys in the BI domain.
getBIServiceInstance	This command gets service instance details for a given service instance key.
scaleOutBIServiceInstance	This command scales-out the Service Instance(s) onto the new computer, ensuring that the service instance is available on the specified computer within the BI domain. This command is only used in advanced cases.
export_service_instance	This command exports a service instance to a given export directory in the form of BAR file.
import_bar	This command imports an already exported bar (service instance) as a customization to a given environment.

Command	Description
refreshServiceInstance	This command refreshes certain aspects of a service instance serviceKey that are inherited from the BI domain. For example, if a new module or product is added to the BI domain, the permission sets for that module or product will only be made available to the service instance when the service instance is refreshed.
refreshDomainServiceInstances	This command refreshes all the service instances of the domain.
resetServiceInstance	This command resets the given service instance to empty state equivalent to importing the empty BAR file.
Parameters	Description
domainHome	Path to BI domain home. <code>/oraclehome/user_projects/domains/bi</code>
serviceInstanceKey	Key for the service instance to be associated with, or scaled out to. For example: <code>mycompany.facility</code>
machine	Name of the computer to which you are scaling out. For example: <code>machine=mycompany.example.com</code>
port	The port number to use on the scaled out computer. For example: <code>port=9768</code>
monitorPort	Name of the monitor port to use on the scaled out computer. For example: <code>portMonitor=9502</code>

Assumptions for all WLST commands against BI Service Instances and BAR files:

- You must have file system (offline) permissions.
- You run the WLST commands offline.
- You must start your system after making changes to service instances through WLST.

The following scripts are reserved for future use:

- `ORACLE_HOME/user_projects/domains/bi/bitools/bin/create_bi_service_instance.sh`
- `ORACLE_HOME/user_projects/domains/bi/bitools/bin/delete_bi_service_instance.sh`

listBIServiceInstances

This command enables you to list all Service Instances.

1. Start the WLST command line scripting tool.

See [Using the WebLogic Scripting Tool \(WLST\)](#).

2. Enter command, for example:

```
listBIServiceInstances(domainHome)
```

Where domainHome is the path to BI domain home.

For example:

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

3. The command returns a list of Service Instance Keys

See [getBIServiceInstance](#).

getBIServiceInstance

This command displays service instance details.

Information includes but is not limited to:

- Metadata configuration: application association and customizations.
- Component details.
- Description.

1. Enter the following command to fetch the specified Service Instance details:

```
getBIServiceInstance(domainHome, serviceInstanceKey)
```

For example:

```
getBIServiceInstance('/oraclehome/user_projects/domains/bi',  
'mycompany.facility')
```

2. The command returns a service instance object containing details including, but not limited to:

- Service instance key.
- Description.
- List of components.
- List of application modules.

scaleOutBIServiceInstance

This command scales-out the service instances onto a new computer to make the service instance available in the domain.

Use this command if a service instance is created after availability is increased. See [Managing Availability in Oracle Business Intelligence \(Horizontally Scaling\)](#).

```
scaleOutBIServiceInstance(domainHome, serviceInstanceKey, machine, port=None,  
portMonitor=None)
```

This command returns a Service Instance object containing details.

Assumptions:

- All ports will be allocated from the default BI port range, unless you provide them.
- Cluster Controller, Scheduler and BI Server mastership is unchanged.
- You must start new components, noting that Administration Server and Node Managers must be started first if offline. See [Starting Oracle Business Intelligence Component Processes in a Domain](#).

The following table lists the parameters for this command.

Parameters	Description
domainHome	Path to BI domain home.
serviceInstanceKey	Key for the service instance to be scaled out.
machine	Name of the computer to which you are scaling out the service instance.

For example,

```
scaleOutBIServiceInstance('/oraclehome/user_projects/domains/
bi','mycompany.facility', 'example.com')
```

Post Conditions:

- New component(s) are created.
- New ports(s) are allocated.
- New Service Instance is registered.

export_service_instance

This command exports metadata in the repository from a specified BAR file into a target service instance.

You can then import the BAR file into another environment or within the same environment. Details of these optional parameters are given in the table below.

```
export_service_instance(domainHome serviceInstanceKey workDir, exportDir,
applicationModuleName, applicationModuleDesc, applicationModuleVersion,
includeCatalogRuntimeInfo, includeCredentials)
```

Parameters	Description
domainHome	Path to BI domain home.
serviceInstanceKey	Key for the service instance.
workDir	Work directory for the run.
exportDir	Directory where you want to export the service instance to create the BAR file.
applicationModuleName	Reserved for future use.
applicationModuleDesc	Reserved for future use.
applicationModuleVersion	Reserved for future use.

Parameters	Description
includeCatalogRuntimeInfo	Optional. Default is False. If this parameter is set to True, catalog runtime details (such as the user folder name) are included in the export of the service instance. If this parameter is set to False, catalog runtime details aren't included.
includeCredentials	Optional. Default is None. The password to encrypt the contents of the exported metadata repository. If you don't specify a value for this parameter, connection credentials aren't exported.

For example,

```
export_service_instance( '/oraclehome/user_projects/domains/
bi','mycompany.facility', '/workDir', '/scratch/exportDir')

export_service_instance('/oraclehome/user_projects/domains/bi/', 'mycompany.dev', '/
scratch/workDir', '/scratch/exportDir', 'mycompany.dev.test', 'mycompany dev test',
'11.1.1.0')

export_service_instance('/oraclehome/user_projects/domains/bi/', 'mycompany.dev', '/
scratch/workDir', '/scratch/exportDir', 'mycompany.dev.test', 'mycompany dev test',
'11.1.1.0', true, 'Test123')

export_service_instance('oraclehome/user_projects/domains/bi/', 'mycompany.dev', '/
scratch/workDir', '/scratch/exportDir', 'mycompany.dev.test', 'mycompany dev test')

export_service_instance('/oraclehome/user_projects/domains/
bi/', 'mycompanyexample.dev', '/scratch/workDir', '/scratch/exportDir', None ,
'mycompany dev test')
```

 **Note:**

Running this command will place your system in maintenance mode and write access will be limited for the duration of the process.

import_bar

This command imports metadata in the repository from a specified BAR file into a target service instance.

```
import_bar(domainHome, serviceInstanceKey, barLocation, importRpd, importWebcat,
importJazn, includeCredentials)
```

The table lists the parameters for this command.

Parameters	Description
domainHome	Path to BI domain home.
serviceInstanceKey	Key for the service instance.
barLocation	Absolute path of the BAR file for the imported service instance.

Parameters	Description
importRpd	Optional. Default is True. If you set this parameter to False, the command doesn't import the metadata from the repository.
importWebcat	Optional. Default is True. If you set this parameter to False, the command doesn't import the objects from the catalog.
importJazn	Optional. Default is True. If you set this parameter to False, the command doesn't import the Jazn.
includeCredentials	Optional. Password for decrypting the content imported the repository.

For example,

```
import_bar( '/scratch/mydir/oraclehome/user_projects/domains/
bi','mycompany.facility', '/scratch/exportDir/mycompany.finance.bar')

import_bar('/scratch/mydir/oraclehome/user_projects/domains/bi', 'demosvc1', '/
scratch/mydir/oraclehome/bi/bifoundation/samples/sampleapplite/test/
SampleAppLite1.bar', true, false, false, 'Test123')

import_bar('/scratch/mydir/oraclehome/user_projects/domains/bi', 'demosvc1', '/
scratch/mydir/orahome/bi/bifoundation/samples/sampleapplite/test/
SampleAppLite1.bar', importRPD=true,includeCredentials='Test123')
```

Note:

When you import a 12.2.1.2.0 or later BAR file into an earlier Oracle BI EE release such as 12.2.1.1.0 and 12.2.1.0.0 and the BAR file contains dataset metadata (for example an Excel spreadsheet) that isn't supported before 12.2.1.2.0, the dataset metadata isn't imported and the metadata that isn't imported also isn't logged.

- When you import a later version BAR file that contains dataset metadata (for example an Excel spreadsheet called sales.xls) into an earlier Oracle BI EE instance that doesn't support dataset metadata, the import process silently skips the dataset metadata content in the BAR file.
- The log file for the service instance contains information on the metadata that's imported. The log file doesn't provide information on the metadata that's ignored as part of the import operation.

refreshServiceInstance

This command refreshes certain aspects of a service instance that are inherited from the BI domain.

For example, if a new module or product is added to the BI domain, the permission sets for that module or product will only be made available to the service instance when the service instance is refreshed.

```
refreshServiceInstance(domainHome, serviceKey)
```

The following table lists the parameters for this command.

Parameters	Description
domainHome	Path to BI domain home.
serviceKey	Key for the service instance. It is a unique identifier for the service instance.

For example,

```
refreshServiceInstance('/oraclehome/user_projects/domains/bi', 'mycompany.finance')
```

refreshDomainServiceInstances

This command refreshes all the service instances of the BI domain.

```
refreshDomainServiceInstances(domainHome)
```

The following table lists the parameters for this command.

Parameters	Description
domainHome	Path to BI domain home.

For example,

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

resetServiceInstance

This command removes all the customizations in a given service instance (similar to an empty BAR state).

```
resetServiceInstance(domainHome, serviceKey)
```

The following table lists the parameters for this command.

Parameters	Description
domainHome	Path to the BI domain home. For example, /oraclehome/user_projects/domains/bi.
serviceKey	Key for the service instance. It is a unique identifier for a service instance. For example, 'ssi'.

This example removes customizations using the resetServiceInstance command.

1. Configure Oracle Business Intelligence with an empty service instance (the same as using an empty BAR file).

Note:

The empty BAR file is here:

```
$BI_PRODUCT_HOME/bi/bifoundation/admin/provisioning/  
oracle.bi.application.empty.bar
```

2. Deploy SampleAppLite to the BI domain and associate it with the service instance.
3. Make some customizations on top of the base BAR.
4. Use the `resetServiceInstance` command.

For example,

```
resetServiceInstance('/oraclehome/user_projects/domains/bi', 'ssi')
```

This removes the customizations created in step 3 and returns the service instance to the state of step 2.

10

Configuring Connections to External Systems

This chapter describes how to configure connections to systems that are external to Oracle Business Intelligence.

This chapter includes the following sections:

- [Configuring Email and Agents](#)
- [Configuring for Actions with the Action Framework](#)
- [Configuring Connections to the Marketing Content Server](#)
- [Configuring Connections to Data Sources](#)

As part of the process of configuring connections to external systems, you can configure a database for the Oracle BI Scheduler. See *Using Fusion Middleware Control to Configure a Database for the Oracle BI Scheduler* in *Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*.

Configuring Email and Agents

You can use Fusion Middleware Control to configure common email settings that are used by agents.

Advanced configuration settings are described in [Configuring and Managing Agents](#).

Using Fusion Middleware Control to Configure Oracle BI 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 are familiar with the information in .

1. Go to the Business Intelligence 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. The default is Oracle Business Intelligence.

- **Email address of sender**

This option specifies the email address on the SMTP Server that is used as the sender's reply-to address for all mail sent from Oracle BI 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 need not 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**
 - **SSL certificate verification depth**
 - **SSL cipher list**
5. Click **Apply**, then click **Activate Changes**.
 6. Return to the Business Intelligence Overview page and click **Restart**.

For information about advanced configuring settings for agents, see [Configuring and Managing Agents](#).

For information about corresponding configuration file elements, see [Mapping User Interface Labels with Configuration File Elements](#).

Configuring for Actions with the Action Framework

Users can create *actions* in the Oracle BI Presentation Services user interface.

An action is an operation or process that can be invoked explicitly by a user clicking an action link. Actions can also be invoked automatically, as the final step of an agent.

You can configure for the use of actions in your organization. For a comprehensive discussion of how to use the Action Framework to enable actions for external systems, including a complete description of each configuration setting and detailed examples, see *Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Configuring Connections to the Marketing Content Server

Oracle Marketing Segmentation handles segmentation, which involves dividing a target audience into different segments given different criteria based on the subject areas.

When a segment is ready, users create lists of the contacts and accounts that satisfy the criteria for the segment. Users then specify whether to store the generated lists on the file system, in a database, or on a specified Content Server.

For users to store the lists on a Content Server, you as the administrator must configure the connection to the Content Server by specifying the appropriate URL and other values in the `instanceconfig.xml` file.

Use various elements in the `instanceconfig.xml` file to configure settings for the connection to the Marketing Content Server.

1. Open the `instanceconfig.xml` file for editing, at:

```
BI_DOMAIN/config/fmwconfig/biconfig/OBIPS
```

2. Locate the section in which you must add the elements that are described in the following list:

- URL: Specifies the address of the content server machine.
- SocketTimeoutSec: Specifies the number of seconds that the socket waits for the Content Server to respond while transferring records. The default value is 60. There is no minimum or maximum value.
- FileSizeMB: Specifies the size in megabytes of files that are generated during ListGeneration and inserted into the Content Server. The default is 10. The minimum size is 1MB and the maximum size is 50MB.

3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Marketing>
    <ContentServer>
      <URL>myhost.com:6666/st1b2rep/idcplg</URL>
      <SocketTimeoutSec>120</SocketTimeoutSec>
      <FileSizeMB>5</FileSizeMB>
    </ContentServer>
  </Marketing>
</ServerInstance>
```

You cannot specify values for user name and password in the `instanceconfig.xml` file. Instead you specify values 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 Oracle Business Intelligence.

Configuring Connections to Data Sources

Connections to data sources are defined in the Oracle BI repository. Repository developers use the Administration Tool to configure data source connections by importing metadata and configuring connection pools.

See *Importing Metadata and Working with Data Sources* in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

You might need to update connection pool information in the repository 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 *XML Schema Reference for Oracle Business Intelligence Enterprise Edition*.

11

Configuring Presentation Setting Defaults

This chapter describes how to change default presentation settings in Oracle Business Intelligence that administrators commonly change using Fusion Middleware Control. Advanced configuration settings are described in [Manually Changing Presentation Settings](#).

Using 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 [Using Fusion Middleware Control](#).

1. Go to the Business Intelligence 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 Business Intelligence Overview page and click **Restart**.

For information about advanced configuring settings for analyses and dashboards, see [Configuring and Managing Analyses and Dashboards](#).

For information about corresponding configuration file elements, see [Mapping User Interface Labels with Configuration File Elements](#).

12

Configuring Mapping and Spatial Information

This chapter describes how to configure mapping and spatial information for map views in Oracle BI Administration Tool.

When you install Oracle BI 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.

This chapter includes the following sections:

- [What Are the System Requirements for Map Views?](#)
- [Hardware Sizing and Deployment Strategy for Maps](#)
- [Sample Spatial Dataset for use with Map Views](#)
- [Administering Maps](#)

See [Configuring Advanced Options for Mapping 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 BI EE. MapViewer is installed as part of Oracle BI EE and deployed in the same domain as Oracle BI EE 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 [Configuring MapViewer to Support Map Views](#).

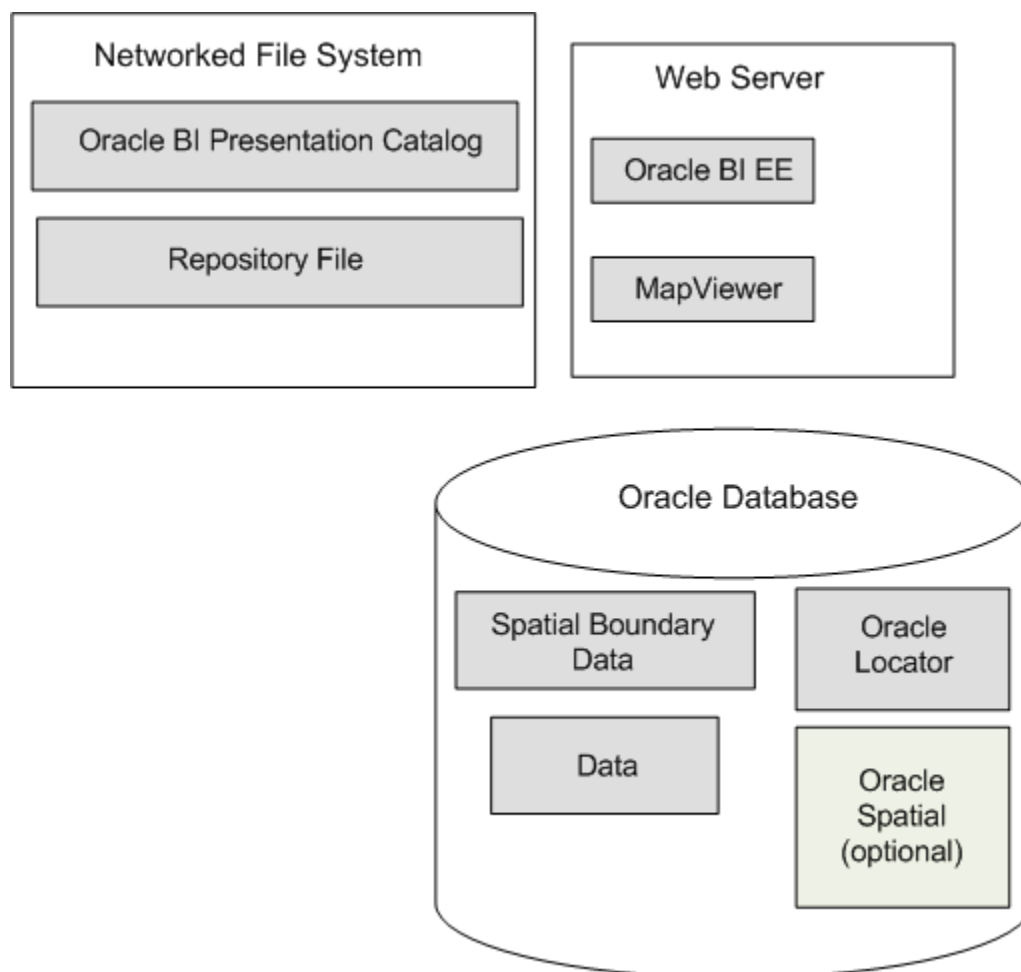
- Spatial boundary data. NAVTEQ is one provider of this data to Oracle customers, which can be downloaded from the Oracle Technology Network. This spatial data and any other spatial metadata, including themes and styles, must be stored in an Oracle Database to be accessed by Oracle MapViewer for display in map views.
- Hosted maps. In Oracle BI EE, 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 10g 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 Business Intelligence* 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 BI EE data and spatial data, which can be stored in a file system, in the Oracle BI Presentation Catalog.

The illustration shows the default architecture for map views when Oracle BI EE is installed. You can store the data either in an Oracle Database or in other databases that Oracle BI EE supports. See [Configuring 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 [Administering Maps](#).

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.

Sample Spatial Dataset for use with Map Views

Learn about locating and setting up setting up the sample MapViewer spatial dataset, and how to obtain a full spatial dataset.

Information About the Sample (and Full) Spatial Dataset

A sample spatial dataset is included in the Sample Application when installed with Oracle Business Intelligence. It is also available from OTN at [Oracle Business Intelligence Enterprise Edition Samples](#).

The sample spatial dataset is included in the OBIEE_NAVTEQ schema of the Oracle Database in the SampleApp image. You can download a standalone export of the OBIEE_NAVTEQ schema (obiee_navteq.dmp) from the SampleApp archives page [OBIEE SampleApp Archives](#) using the link **NAVTEQ Data Bundle for OBIEE** found under OBIEE 11.1.1.3 - Sample Application (Build 825).

Instructions for importing the data bundle are included in "Loading MapViewer Content" in the document accessed from the link **Documentation Downloads**. **Note:** The import process requires SQL scripts in the setup files available from the link **Sample Application - Setup Files**.

The MapViewer sample dataset contains the following: Worldwide Country boundaries, States for US, Canada, Australia, US Counties, major cities of the world, US Highways, street level detail data from 2012 for San Francisco, London, and Sydney only.

The Sample dataset terms of use are here: [Oracle end-user terms](#)

To obtain the full spacial dataset contact NAVTEQ using this URL:

[HERE: Mapping solutions for Oracle developers](#)

Administering Maps

Before content designers can create map views, you as the administrator must specify layers and maps and configure the metadata.

This section discusses the following:

- [Working with Maps and Layers](#)
- [Administration Page Functions](#)
- [Administering Maps Using Administration Pages](#)
- [Handling the Translation of Layers in Maps](#)

Working with Maps and Layers

Maps can have one or more layers with which to work.

The first step is to select the layers for use on maps. An administrator has configured layers using the Map Builder tool of Oracle Spatial. You next select at least one map from a list of those that an administrator has configured using the Map Builder tool of Oracle Spatial. This map becomes the background on which the layers are applied. You can optionally specify images for use with map formats. This section provides the following information about maps and layers:

- [Associating Layers with Columns](#)
- [Ordering Layers on Maps](#)
- [Changes to Spatial Metadata Require Restart](#)

Associating Layers with Columns

After selecting layers and maps, you can associate certain layers with columns in the Oracle Business Intelligence subject area folders.

If the association between a column and a layer is incorrect, then the layer cannot be displayed correctly on the map. The association ensures that shape definitions can be found in the database for the column values during map rendering. You must ensure that a shape geometry definition exists for each column value in the database. If a shape geometry definition does not exist for a particular column value, then the shape cannot be shown on the map and might inhibit user interactions on the map.

Shape lookup is based on the column values, and column-to-layer mapping is independent of locale or language. Therefore, you must ensure that the spatial column that is being associated with a layer does not itself have values that are affected by locale or language. To ensure this association, do one of the following:

- Model spatial columns as double columns in the business modeling layer, which is the recommended method.
- Create special spatial columns that have values that do not change across locale or language. You must ensure that content designers are not confused seeing these special columns in the Subject Areas pane when working with analyses.

If you decide to use double columns, then see *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for more information on double columns. The advantages of using double columns are the following:

- You can provide code values (that is, descriptor IDs) for each shape definition and at the same time show display values (that is, descriptor values) according to locale or language.
- Code values are passed only as layer key values for lookup.
- You eliminate the need for the complex joining of various columns to uniquely locate the shape. For example, a layer geometry table might contain multiple cities with the name of London. To uniquely distinguish these cities, you might use a pattern of Country_State_City, as in US_Kansas_London or Canada_Ontario_London. The problem with this approach is that it might require three separate columns to be grouped, joined by a delimiter (such as an underscore), and associated with a layer. This association requires the content designer to select three columns (Country, State, City) in the criteria to create a single layer.

A BI layer can be associated with multiple columns in multiple subject areas. You must ensure that a layer is associated with at least one spatial column. If the layer association is missing, then the map might not be updated when a user drills on the mapped BI column. You can also have *feature layers*, which are layers that are not associated with a BI column.

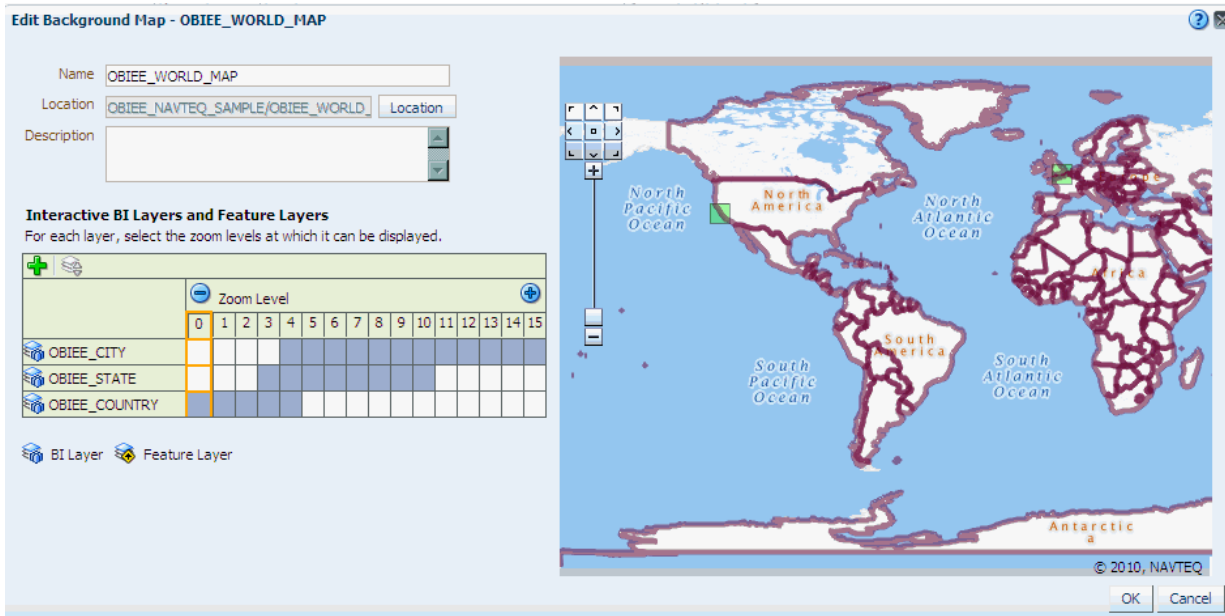
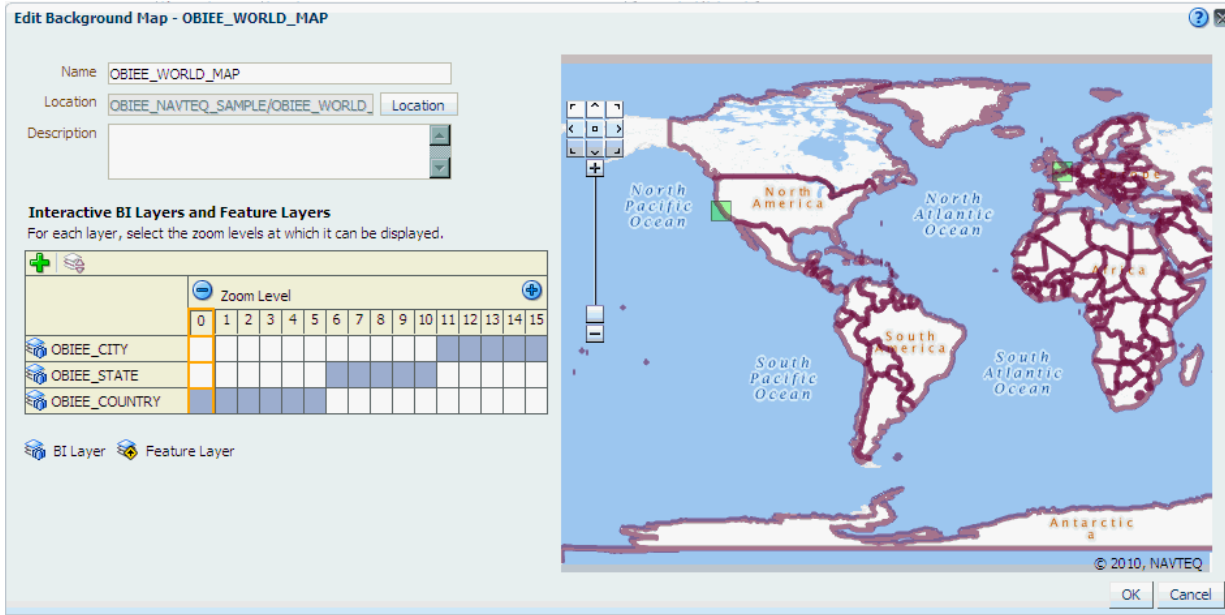
Ordering Layers on Maps

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. Ensure the following:

- That you sort the layers (by clicking the sort icon) before closing the dialog.
- 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.

Layer ordering becomes irrelevant when the zoom ranges of layers do not 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 are not hidden by the aggregated layers during drilling or zooming operations.



The layer ordering at zoom level 4 is unclear, because multiple layers can be shown. Because of the layer order that is specified in the Interactive BI Layers section of the dialog, the map renders with the following order at zoom level 4: Country, State, and City (reading from bottom to top).

If the layers are not ordered properly in the dialog, then you can re-order layers by clicking the **Move Down** and **Move Up** buttons on the zoom grid of the dialog, or by clicking the **Sort Layers by Zoom Level** button.

See *User's Guide for Oracle Business Intelligence Enterprise Edition* for more information about layers.

Example 12-1 World Map with Three Layers

Suppose that a world map has three layers (Country, State, and City) and 15 zoom levels defined on it. The Country layer has a minimum and maximum zoom range of 0-5, the State layer range is 6-10, and the City layer range is 11-15. As the user navigates from the minimum to the maximum zoom level on the map, the layer order (also known as the visual order) is displayed as Country, State, and City.

The first image above shows the Edit Background Map dialog with the Interactive BI Layers section specified for this example. Reading from bottom to top, you see the layer order in that section as Country, State, and City.

Example 12-2 World Map with Common Levels

Consider the same world map with layers that have common zoom ranges. Suppose that the user zooms to level 4 on the map that corresponds to the Edit Background Map dialog that is shown in the second image above. Notice that all three layers include zoom level 4.

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.

Administration Page Functions

The Oracle BI Server Administration page provides the **Manage Map Data** link.

This link displays the Manage Map Data page, where you can manage the logical and display versions of the data from various physical data sources. This defines the layers that content designers use when creating map views. The data that is available for managing maps and data is stored in Oracle Database as part of MapViewer.

Using this page, you provide:

- Logical names to prevent any existing BI column mappings and map analyses from breaking because of changes to the physical data or to the data source.
- Display names so that the geographic data is meaningful to end users.

Administering Maps Using Administration Pages

This section assumes that you installed Oracle BI Enterprise Edition and that Oracle MapViewer was automatically configured and deployed.

For information on MapViewer, see *Map Visualization Developer's Guide*.

1. Sign in to Oracle Business Intelligence.
Ensure that you have been granted the Access to Administration and Manage Map Data privileges.
2. In the global header, click **Administration**.
3. Click the **Manage Map Data** link to display the Manage Map Data page.

4. Click the Layers tab.
5. Click the **Import Layers** button to display the Import Layers dialog.
6. In the dialog, select the connection and the layers that are needed for zooming and drilling. (This tab is not prepopulated by other selections on the Administration pages. You can use any layers or images with any map.)

Click **OK** when you have finished selecting layers that are appropriate for the subject area with which you are working.
7. Back on the Layers tab, select a layer, then click the **Edit Layer** button to display the Edit Layer dialog in which you associate layers with attribute columns so that you can display BI data in the map view.

Click **OK** when you have finished editing the layer.

You use this tab to associate layers with BI data. If you use the City column in multiple subject areas, then you must associate it with a layer for each subject area.
8. Click the Background Maps tab, then click the **Import Background Maps** button to display the Import Background Map dialog.
9. In the dialog, select the connection and the main maps to use.

The connection that you select for the main map can be different from the connection for the layers or images.

Click **OK** when you have finished selecting main maps.
10. Back on the Background Maps tab, select a map, then click the **Edit Background Map** button to display the Edit Background Map dialog in which you name the map and specify the order of layers and their zoom levels.

Click **OK** when you have finished editing the map.
11. Optionally, click the Images tab, then click the **Import Images** button to display the Import Images dialog. You can import images if you plan to use them as a format on maps.
12. In the dialog, select the connection and the images to use.

Click **OK** when you have finished selecting images.
13. Click **Back** when you have finished working with the Administration page. Your changes are saved automatically.

After you have specified background maps, layers, and zoom levels, MapViewer creates a static image for a map using this information. Then MapViewer sends that image for rendering in the browser for use by content designers and end users in map views.

Handling the Translation of Layers in Maps

You can use the functionality of MapViewer to label the features of a theme (called a layer for maps in Oracle BI EE) using a specific language or locale.

To configure these translated labels for maps, use the information that is provided in *Map Visualization Developer's Guide*.

13

Configuring Time Zones

This chapter describes how to configure time zones for Oracle Business Intelligence. This chapter includes the following sections:

- [Why and Where Are Time Zones Used?](#)
- [Setting Time Zones](#)
- [What Is the Precedence Order for Time Zones?](#)
- [Where Are Time Zone Specifications Stored?](#)
- [Description of Time Zone Settings](#)
- [Example: Configuration File Settings for Specifying the Time Zone](#)

Why and Where Are Time Zones Used?

Time zones are used throughout Oracle Business Intelligence 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	<p>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 Business Intelligence. 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.</p> <p>If you make no time zone settings and if a user does not 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.</p> <p>For information about how users specify their preferred time zones, see <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i>.</p>
Data from the database	<p>The administrator specifies the time zone for the data that is retrieved from the database.</p> <p>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.</p>

Type	Description
Content that is displayed in Oracle Business Intelligence	Users who create analyses can specify the time zone that is 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: <ul style="list-style-type: none"> • The scheduled time of agents. • The generated time of alerts or analyses. • The time on which objects in the Oracle BI Presentation Catalog are created, modified, and accessed.
Log files	Log files contain time stamps for various activities.

Setting Time Zones

You can override the user's time zone selection with one that is consistent for all users.

Use the following procedure to set time zones for users.

1. Determine the time zone that is 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 *User's Guide for Oracle Business Intelligence Enterprise Edition* 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 is displayed in the columns of their analyses.
 - b. Use the Time Zone dialog to set the time zone that is 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	<ol style="list-style-type: none"> 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	<ol style="list-style-type: none"> 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)	<ol style="list-style-type: none"> 1. User-Preferred Time Zone 2. The time zone for Oracle BI Presentation Services.
Log file information	<ol style="list-style-type: none"> 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*.

- 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 do not 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 & 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`, cannot 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 Oracle BI Administration Tool.

See *Working with Repository Variables in Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Element	Session Variable	Description	Value
DefaultDataOffset	DATA_TZ	<p>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.</p> <p>If you do not set this option, then no time zone conversion occurs because the value is "unknown".</p> <p>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.</p>	<p>An offset that indicates the number of hours away from GMT time. For example: "GMT-05:00" or "-300", which means minus 5 hours.</p>
DefaultDataDisplay	DATA_DISPLAY_TZ	<p>Specifies the time zone to use for displaying data.</p> <p>If you do not set this option, then the value is the User-Preferred Time Zone.</p>	<p>One of the time zones that are specified in the TimeZones.xml file.</p> <p>See Specifying Time Zone Values</p>
DefaultUserPreferred	TIMEZONE	<p>Specifies the users' default preferred time zone before they select their own in the My Account dialog.</p> <p>If you do not set this option, then the value is the local time zone from Oracle BI Presentation Services.</p>	<p>One of the time zones that are specified in the TimeZones.xml file.</p> <p>See Specifying Time Zone Values</p>
Logging	Not applicable	<p>The time zone of the time stamps that appear in log files that are generated by Presentation Services.</p> <p>If you do not set this option, then the value is the local time zone from Presentation Services</p>	<p>One of the time zones that are specified in the TimeZones.xml file.</p> <p>See Specifying Time Zone Values</p>
TimeZone	Not applicable	<p>The parent element for the elements that modify the preferred time zone. A child of the ServerInstance element.</p>	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 & Canada); Tijuana</Logging>
  <DefaultUserPreferred>(GMT-08:00) Pacific Time (US & Canada); Tijuana</
DefaultUserPreferred>
```

```
<DefaultDataDisplay>(GMT-06:00) Central Time (US & Canada); Tijuana</  
DefaultDataDisplay>  
  
</TimeZone>
```


Localizing Oracle Business Intelligence

This chapter describes how to configure Oracle Business Intelligence 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.

This chapter includes the following sections:

- [What Is Localization?](#)
- [Localizing Oracle BI PresentationServices](#)
- [Setting the Current Locale in Catalog Manager](#)
- [Setting the Current Locale in the Oracle BI Server](#)
- [Localizing Metadata Names in the Repository](#)
- [Supporting Multilingual Data](#)

What Is Localization?

Localization refers to the process of adapting the Oracle Business Intelligence 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.

For information about supported languages, see [System Requirements and Certification](#).

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 Business Intelligence are translated into languages other than English:

- Installer
- Web user interface
- Job Manager interface of the Oracle BI Scheduler
- Catalog Manager
- Oracle BI Presentation Services messages:
 - error
 - warning
 - information
- Oracle BI Server functions:

- DayName
- MonthName

 **Note:**

If a query is issued using the DayName or MonthName function, but the function is not shipped to a back-end database, then the day or month name is returned in the localized language but the column name remains in English (or might be affected by other localization controls). As an example of this situation, 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:
 - nqserver.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 (Oracle BI Presentation Catalog)
 - Presentation table and column names (.rpd file)
- Oracle BI Administration Tool interface
- ODBC setup

The following list outlines which components of Oracle Business Intelligence are not localized:

- ODBC client tools:
 - nqcmd (UNIX)
 - 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 for Localizing Oracle Business Intelligence Components

Localizing the system components for international use requires several steps.

As the administrator, you perform various tasks to localize the components of Oracle Business Intelligence, as described in the following sections:

- [Localizing Oracle BI PresentationServices](#)

- [Setting the Current Locale in the Oracle BI Server](#)
- [Localizing Metadata Names in the Repository](#)
- [Supporting Multilingual Data](#)

Localizing Oracle BI PresentationServices

As the administrator, you perform various tasks to localize Oracle BI Presentation Services.

Topics include:

- [Localizing the User Interface for Oracle BI PresentationServices](#)
- [Localizing Oracle BI Presentation Catalog Captions](#)
- [Tip for Arabic and Hebrew in Mozilla Firefox Browsers](#)

Localizing the User Interface for Oracle BI PresentationServices

You can localize the user interface for Oracle BI Presentation Services, if your users speak languages other than English.

Users can select a language on the sign-in page for Oracle BI EE, 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 Oracle BI Presentation Services user interface, WEBLANGUAGE is set when a user selects a language on the sign-in page.

Note:

For Oracle BI 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 BI 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 Oracle BI Presentation Services, as described in the following sections:

- [Understanding the Directory Structure for Localizing Presentation Services](#)
- [Localizing Messages for Users' Preferred Currency](#)
- [Specifying the Default Language for the Sign-In Page](#)
- [Configuring the Languages and Locales for the Sign-In Page](#)
- [Specifying the Language in the URL](#)

Understanding the Directory Structure for Localizing Presentation Services

Oracle BI EE 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/service_instances/service1/metadata/content/msgdb/l_xx`

where `SDD` is the Singleton Data Directory, and `_xx` indicates the language extension. See [Key Directories in Oracle Business Intelligence](#). 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.

Localizing Messages for Users' Preferred Currency

You can localize the messages that are associated with a preferred currency.

See [Defining User-Preferred Currency Options Using a Static Mapping](#) for information on working with users' preferred currencies.

1. Go to the `SDD/service_instances/service1/metadata/content/msgdb/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 in Oracle Business Intelligence](#).

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 BI EE, the appropriate localized text is displayed to the user. In this example, the text is `My Currency Text 1`.

Specifying 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. The following procedure explains how to change the language.

 **Note:**

The following procedure uses Internet Explorer 7.0 as an example. If you are using a different browser, then make the necessary substitutions.

1. In Internet Explorer, from the **Tools** menu, select **Internet Options**.
The Internet Options dialog is displayed.
2. Click **Languages**.
The Language Preference dialog is displayed.
Installed languages are displayed in the Language list. The language at the top of the list is used as the default language.
3. If the desired language is not installed on the browser, then add it.
4. Use the **Move Up** and **Move Down** buttons to position the desired language at the top of the list.
5. Restart the browser and sign into Presentation Services.
The default language likely matches the language in the browser's Language list.

 **Note:**

If a user does not select a different language from the list on the sign-in page, then the setting for the User Interface Language in the user's My Account dialog determines the language in which the user interface is displayed.

Configuring the Languages and Locales for the Sign-In Page

You can configure the languages and locales that are available to users on the sign-in page.

This ability is helpful for limiting the number of languages and locales that users can access. You use the `AllowedLanguages` and `AllowedLocales` elements in the `instanceconfig.xml` file to specify the available languages and locales.

Ensure that you are familiar with the information in [Configuration Files](#) before you start.

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 following elements:
 - `AllowedLanguages` — Specifies the languages that are available for selection, as a comma-delimited list. You can specify a list of the following identifiers, which are ISO 639 language codes:
 ar es da de el en es fi fr he hr hu it ja ko nl no pl pt pt-BR ro ru sk sv th tr zh-CN zh-TW
 - `AllowedLocales` — Specifies the locales that are available for selection, as a comma-delimited list. You can specify any definition from the `localedefinitions.xml` file in the `ORACLE_HOME/bi/bifoundation/web/display` directory. You specify the locales using ISO 639 language codes followed by ISO 3166 country codes. Examples include `fr-fr` and `fr-ca`.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:


```
<ServerInstance>
  <Localization>
    <AllowedLanguages>en,fr,pt-BR</AllowedLanguages>
    <AllowedLocales>en-ca,en-us</AllowedLocales>
  </Localization>
</ServerInstance>
```
4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Specifying the Scaling of 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>
</localeDefinition>
```

```

        <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.

See Editing Performance Tile Views in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

Specifying the Language in the URL

When users start Oracle BI EE 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 Business Intelligence. 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

Name	Type	Path Argument Value	Append	Sequence #
nQPassword	Command	UseLoginPassword	Y	4
PostRequest	Command	PostRequest	Y	5
Lang	Profile Attribute	LanguageCode	Y	6

Localizing Oracle BI Presentation Catalog Captions

The Oracle BI Presentation Catalog stores objects that users create, such as analyses and dashboards. Text strings hold the names and descriptions of these objects.

If you must localize text strings for the objects, then you can export the text strings from the catalog so that they can be translated. You then expose the strings when translation is complete.

This section describes the steps in the process of localizing captions:

- [Step 1: Understanding the Export Process](#)
- [Step 2: Exporting Text Strings in the Catalog](#)
- [Step 3: Editing Exported Strings in XML Files](#)
- [Step 4: Handling Duplicate Exported Text Strings](#)
- [Step 5: Exposing Text Strings in the Catalog](#)

Step 1: Understanding the Export Process

The export process requires a series of steps that must be followed in order.

The export process creates one XML file for every first-level subfolder in the shared folder, in the format *foldername* captions.xml, where *foldername* is the name of the subfolder in the shared folder. Each XML file contains the text strings for all content in the corresponding first-level folder and its subfolders.

For example, if the shared folder in the Presentation Catalog contains the first-level folders Marketing, Service, and Sales, then the export process creates three XML files:

- marketingcaptions.xml
- salescaptions.xml
- servicecaptions.xml

After the content is translated, you place these folders in their corresponding location in the following directory:

```
SDD/service_instances/service1/metadata/content/msgdb/l_xx/
captions
```

Where SDD is the Singleton Data Directory for example, *DOMAIN_HOME*/bidata (see [Key Directories in Oracle Business Intelligence](#)).

The export process not only generates new XML files, but the process also modifies the catalog, inserting the appropriate message ID for each object. Presentation Services uses those message IDs to locate the newly translated text.

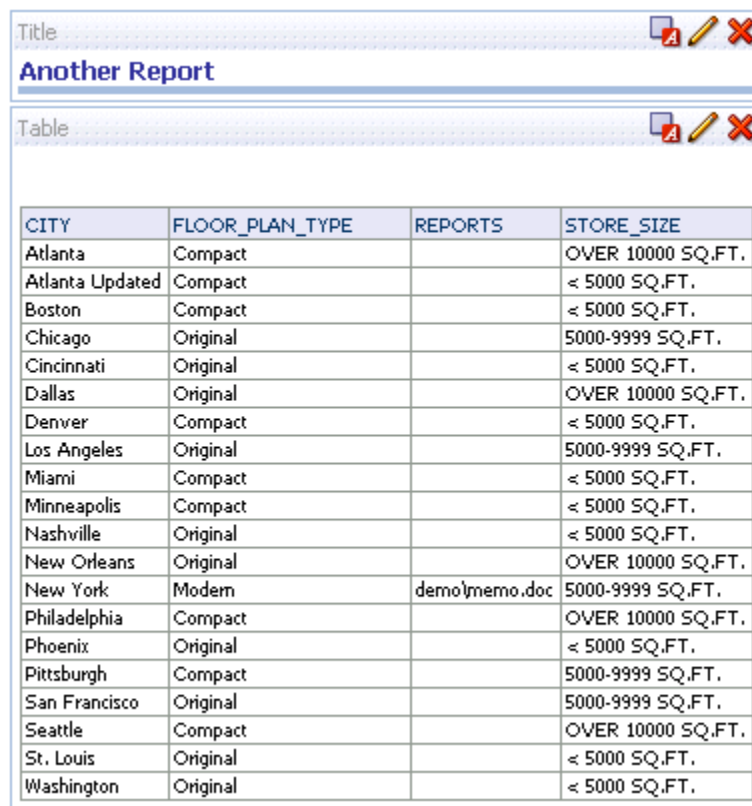
Note that an error may occur when you export a folder whose name includes supplementary (extended Unicode) characters.

Step 2: Exporting Text Strings in the Catalog

The following procedure describes how to export text strings in the catalog.

1. Back up the catalog before exporting from it.
Ensure that you run the export utility against the actual catalog, not against a copy of the catalog, because the export utility changes the properties of the objects in the catalog against which it runs.
2. In Catalog Manager, open the catalog in offline mode, because you might lack permission to modify all objects in online mode.
3. Select the folder that contains the strings to export. The utility runs against the files in that folder and all its subfolders.

For example, the title (Another Report) in the analysis that is shown in the following illustration can be exported for translation.



CITY	FLOOR_PLAN_TYPE	REPORTS	STORE_SIZE
Atlanta	Compact		OVER 10000 SQ.FT.
Atlanta Updated	Compact		< 5000 SQ.FT.
Boston	Compact		< 5000 SQ.FT.
Chicago	Original		5000-9999 SQ.FT.
Cincinnati	Original		< 5000 SQ.FT.
Dallas	Original		OVER 10000 SQ.FT.
Denver	Compact		< 5000 SQ.FT.
Los Angeles	Original		5000-9999 SQ.FT.
Miami	Compact		< 5000 SQ.FT.
Minneapolis	Compact		< 5000 SQ.FT.
Nashville	Original		< 5000 SQ.FT.
New Orleans	Original		OVER 10000 SQ.FT.
New York	Modem	demo\memo.doc	5000-9999 SQ.FT.
Philadelphia	Compact		OVER 10000 SQ.FT.
Phoenix	Original		< 5000 SQ.FT.
Pittsburgh	Compact		5000-9999 SQ.FT.
San Francisco	Original		5000-9999 SQ.FT.
Seattle	Compact		OVER 10000 SQ.FT.
St. Louis	Original		< 5000 SQ.FT.
Washington	Original		< 5000 SQ.FT.

4. From the **Tools** menu, select **Export Captions**.
5. Click **Browse** to select the location in which to write the output file, then click **OK**.
6. To export only new text strings and those that have been changed since the last export, select **Only export new or changed strings**.
7. To exclude the Description properties from the export, select **Exclude Descriptions**.

8. Click OK.

The export process might take several minutes.

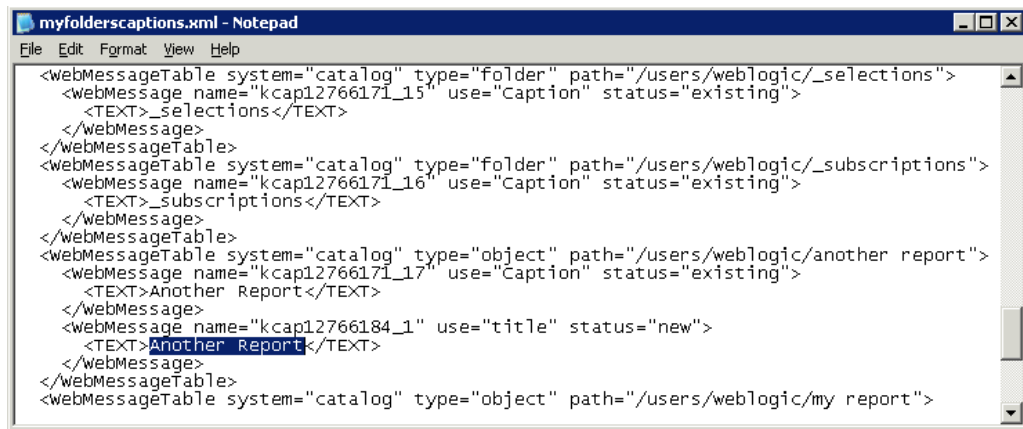
Step 3: Editing Exported Strings in XML Files

When the export process is complete, deliver the output XML file to the localization team.

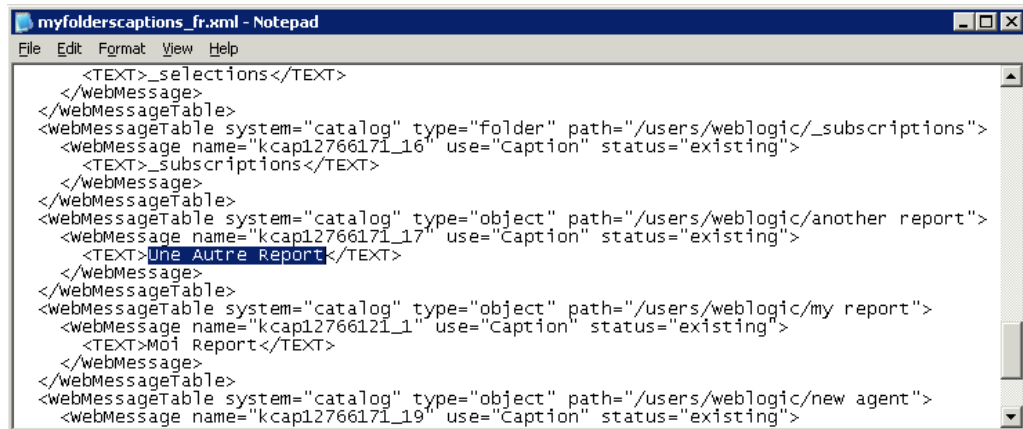
When editing XML files, use an editor that is designed for XML files. Ensure that you follow the encoding that is specified at the top of the XML file and that you escape special characters as appropriate. You and the localization team are responsible for resolving any errors in the translated text strings. Consider that the contents of the catalog are updated whenever objects are added, deleted, or modified.

You can make a copy of every output file for each language to be translated.

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`.



```
myfoldercaptions.xml - Notepad
File Edit Format View Help
<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">
```



```
myfoldercaptions_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">
```

Step 4: Handling Duplicate Exported Text Strings

You might encounter an issue of having duplicate exported text strings from the catalog.

Duplicated exported text strings from the catalog happen when the Export Captions utility is run simultaneously by multiple users or if the same user runs the utility twice in less than one minute. The following procedure describes how to address duplicate captions.

The Export Captions utility is described in [Step 2: Exporting Text Strings in the Catalog](#)

1. Run the Export Captions utility.
2. In Catalog Manager, with the catalog still open in offline mode, select the folder that contains the strings to export.
3. From the **Tools** menu, select **Export Captions**.
4. Click **Browse** to select the location in which to write the output file, then click **OK**.
5. In the "What to do with duplicate captions" section, select one of the following options:
 - **Create unique IDs even for identical strings** — Specifies to create a unique ID for each instance of a string, even if the string is duplicated many times in the catalog. For example, suppose that a catalog includes the "Hello" string 1000 times. Use this option to specify that rather than generating one unique ID and translating the string only once, you want to instead generate 1000 unique IDs and translate the string 1000 times.
 - **No, use the same ID for all identical strings** — Specifies to create an ID for a string and use that same ID for all instances of that string. For example, suppose that a catalog includes the "Hello" string 1000 times. Use this option to specify that you want to generate one unique ID and translate the string only once, instead of generating 1000 unique IDs and translating the string 1000 times.
6. Click **OK**.

Consider the following webmessages.xml file, which contains duplicate captions:

```
<WebMessageTable system="catalog" type="folder" path="/shared/example/A">
  <WebMessage name="kcap12790830_5" use="Caption" status="new">
    <TEXT>A Really Good Report</TEXT>
  </WebMessage>
</WebMessageTable>
<WebMessageTable system="catalog" type="folder" path="/shared/example/B">
  <WebMessage name="kcap12790830_5" use="Caption" status="new">
    <TEXT>I like this report</TEXT>
  </WebMessage>
</WebMessageTable>
<WebMessageTable system="catalog" type="folder" path="/shared/example/Copy of A">
  <WebMessage name="kcap12790830_5" use="Caption" status="new">
    <TEXT>A Really Good Report</TEXT>
  </WebMessage>
</WebMessageTable>
```

In this example file, Object B has an invalid duplicate message ID. Object Copy of A has a valid but duplicate message ID. You can make the following selections in the Export Captions dialog:

- Selecting **Leave alone** makes no changes to the contents of the file.
- Selecting **Remove IDs** generates new and unique IDs for both Object B and Object Copy of A.
- Selecting **Remove texts** generates a new and unique ID for Object B and deletes the WebMessage element for Object Copy of A. While this option generally ensures fewer messages to translate, keep in mind that you now see two objects with the same name in a directory listing of the catalog in Presentation Services and in Catalog Manager.

Step 5: Exposing Text Strings in the Catalog

After you have exported the text strings for the catalog, you must expose them for users.

1. Place the translated XML files into their corresponding location in the following directory and restart Presentation Services:

```
SDD/service_instances/service1/metadata/content/msgdb/l_xx/  
captions
```

where SDD is the singleton data directory and xx is the language extension.

For example:

```
DOMAIN_HOME/bidata/service_instances/service1/metadata/  
content/msgdb/l_en/captions/myfoldercaptions_fr.xml
```

Other examples of language extensions include cs for Czech and de for German.

Note:

The `SDD/service_instances/service1/metadata/content/msgdb/l_xx/captions` directory exists only if Oracle Business Intelligence Applications have been installed. If it does not exist, then you must create it.

Where SDD is the Singleton Data Directory for example,
`DOMAIN_HOME/bidata`.

2. The export process not only generates new XML files, but the process also modifies the catalog, inserting the appropriate message ID for each object. After placing the translated XML files in the directory as specified in Step 1, place the modified catalog in either its default location or in the appropriate location that the system now uses. See [Key Directories in Oracle Business Intelligence](#) for information.
3. Sign into Oracle Business Intelligence and select the appropriate language, such as French.
4. Display the translated content.

For example, display the translated title in an analysis, as shown in the following illustration.

Titre			
Une Autre Report			
Table			
CITY	FLOOR_PLAN_TYPE	REPORTS	STORE_SIZE
Atlanta	Compact		OVER 10000 SQ.FT.
Atlanta Updated	Compact		< 5000 SQ.FT.
Boston	Compact		< 5000 SQ.FT.
Chicago	Original		5000-9999 SQ.FT.
Cincinnati	Original		< 5000 SQ.FT.
Dallas	Original		OVER 10000 SQ.FT.
Denver	Compact		< 5000 SQ.FT.
Los Angeles	Original		5000-9999 SQ.FT.
Miami	Compact		< 5000 SQ.FT.
Minneapolis	Compact		< 5000 SQ.FT.
Nashville	Original		< 5000 SQ.FT.
New Orleans	Original		OVER 10000 SQ.FT.
New York	Modern	demo\memo.doc	5000-9999 SQ.FT.
Philadelphia	Compact		OVER 10000 SQ.FT.
Phoenix	Original		< 5000 SQ.FT.
Pittsburgh	Compact		5000-9999 SQ.FT.
San Francisco	Original		5000-9999 SQ.FT.
Seattle	Compact		OVER 10000 SQ.FT.
St. Louis	Original		< 5000 SQ.FT.
Washington	Original		< 5000 SQ.FT.

To move translated captions from a development environment to a production environment:

- If the caption file:
 - Does *not* exist in the production environment, then simply copy it from the development environment to the production environment.
 - Does exist in the production environment, first make a backup copy of the existing file. Then open the caption file in the production environment in a text editor or XML editing tool and manually (and very carefully) insert the changes that were made in the development environment.

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 are using the Arabic or Hebrew languages, then it is 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.

Setting the Current Locale in Catalog Manager

When you use Catalog Manager, you can specify the locale to use for its user interface elements and for objects in the catalog.

Catalog Manager locales can be the same or different. The user interface elements are available in 10 locales, and the catalog content for certain applications is available in 28 locales.

You can see the user interface elements of Catalog Manager (dialogs, menus, and so on) in any of the 10 locales in which it is available. Certain areas of Catalog Manager, such as data handling, are not yet translated or localized. Catalog Manager uses the following process to decide which locale to display.

1. Check for the setting of the "-nl <locale>" parameter when Catalog Manager is started. You set this parameter as part of the CATMAN_VMARGS variable in the runcat.cmd or runcat.sh file, as shown in the following examples:

```
set CATMAN_VMARGS=-nl fr -vmargs -Xmx1024M -Dosgi.clean=true  
set CATMAN_VMARGS=-nl fr_CA -vmargs -Xmx1024M -Dosgi.clean=true
```

2. Check for the default locale for Java, as specified on your computer.
3. Using the default locale of English (specifically, en_US).

When you start Catalog Manager and open a catalog in online mode, you can select the locale for viewing the contents of the catalog. The locales that are available for selection depend on the following criteria. Catalog Manager uses this selection for subsequent online connections.

- Whether that locale was selected for Presentation Services during the installation process.
- Whether the contents of the catalog have been translated for a specified locale.

If translated files are not available for that locale, then the contents are displayed in the default locale of English (specifically, en_US).

Note that:

- Session errors (such as login failed or session timed out) are displayed in the default locale, not necessarily the locale of the user trying to login or whose session timed out.
- Some strings in the Catalog Manager user interface (such as the string Maximize) are not translated.

See [About Catalog Manager](#).

Setting 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:

- [Setting Locale Parameters on the Oracle BI Server](#)
- [Understanding How the Error Message Language Is Determined](#)
- [Setting the Language for Components of the Oracle BI Server](#)

- [Modifying the Language of the User Interface for the Administration Tool](#)
- [Troubleshooting the Current Locale in the Oracle BI Server](#)
- [Ensuring That Text for Oracle BI Server Utilities is Displayed in the Correct Language](#)
- [Modifying the Metadata Repository When the Underlying Oracle Database NLS_CHARACTERSET Is Not Unicode](#)

Setting 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 Business Intelligence, 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.

Setting the Locale on UNIX 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 UNIX platforms. For example, Chinese uses the setting zh_CN.utf8 on HP-UX or Linux operating systems.

Name strings such as zh_CN.utf8 and fr-FR-UTF-8 are the 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

Locale (Platform-Independent Name)	Name on Solaris	Name on AIX	Name on HP-UX/ Linux
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

Understanding How the Error Message Language Is Determined

For Oracle BI 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 BI EE 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 is not set, then the error message language is based on the error language that is specified in the ODBC Data Source Name (DSN) that is used to access the Oracle BI Presentation Services.

See *Integrator's Guide for Oracle Business Intelligence Enterprise Edition* for information about setting the error message language in the ODBC DSN.

- If an error message language has not been set in the ODBC DSN, then the language that is 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 NQConfig.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 is not set, then error messages are displayed in English.

Note that clients for the Administration Tool and Job Manager do not set the WEBLANGUAGE session variable. Therefore, these clients follow the precedence model starting with the ODBC DSN error message setting.

Setting 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 NQServer.log file, you must set the ORACLE_BI_LANG variable, as described in the following procedure.

1. Open the NQConfig.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 BI system components and the BI Server.

See [Managing Oracle Business Intelligence Processes](#) .

Modifying the Language of the User Interface for the Administration Tool

The user interface of the Oracle BI Administration Tool inherits the language that is specified for the operating system.

For example, if the Windows 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 Windows-based applications such as Notepad and the Administration Tool. The locale that you set in the Windows Control Panel 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 Administration Tool to inherit the language from the operating system. If you must change the language for the user interface of the 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 [Understanding 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 [Localizing Metadata Names in the Repository](#).

Troubleshooting 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:

- [Handling the NLS Locale Not Supported Error Message](#)
- [Setting the Japanese Locale on AIX Systems](#)

Handling the NLS Locale Not Supported Error Message

If you do not have the appropriate locale installed, then the Oracle BI Server does not start, and the `obis1-query.log` file contains the following error:

NLS locale xxx is not 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:

- **UNIX.** Install the locale that is indicated in the table displayed in [Setting the Locale on UNIX Systems](#) for the requested language.
- **Windows.** Add the corresponding language pack using the Regional Settings dialog.

Setting the Japanese Locale on AIX Systems

AIX systems do not always interpret Japanese localization properly.

When using a Japanese localization on an AIX platform, you might discover that the Oracle BI Server does not start. If you encounter this issue, then use the following procedure.

1. Ensure that the `JA_JP.UTF-8` locale is installed. If it is 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.

Ensuring 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 is not displayed in the list, then first change the code page to 65001, which supports UTF-8, using the command `chcp 65001`.

Modifying the Metadata Repository When the Underlying Oracle Database NLS_CHARACTERSET Is Not Unicode

When the NLS_CHARACTERSET in the Oracle Database is set to a non-unicode derived code page, you must configure an additional metadata repository setting to make character sorting consistent between the Oracle Database and the Oracle BI Server.

1. In the Administration Tool, open the metadata repository.
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 repository, and restart the Oracle BI Server.

Localizing Metadata Names in the Repository

You can use the Externalize Strings utility in the 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 repository in the 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 will also appear 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 have 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 repository 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 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 executed first.

6. Save your changes.

Tips:

For information on the language for the Administration Tool, see [Modifying the Language of the User Interface for the 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 *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Supporting Multilingual Data

Oracle BI Server supports several language translations.

This section describes how you can configure the Oracle BI Server to display field information in multiple languages, and contains the following topics:

- [What is Multilingual Data Support?](#)
- [What is Lookup?](#)
- [What is Double Column Support?](#)
- [Designing Translation Lookup Tables in a Multilingual Schema](#)
- [Creating Logical Lookup Tables and Logical Lookup Columns](#)
- [Creating Physical Lookup Tables and Physical Lookup Columns](#)

- [Supporting Multilingual Data in Essbase Through Alias Tables](#)
- [Enabling Lexicographical Sorting](#)

For information about using the Administration Tool, see *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

What is Multilingual Data Support?

Multilingual data support is the ability to display data from database schemas in multiple languages.

Oracle BI Server 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 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 do not allow LOB columns to be used in the `GROUP BY` clause. So, instead of adding the LOB column to the `GROUP BY`, it is 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.

Designing 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

Creating Logical Lookup Tables and Logical Lookup Columns

This section describes creating logical lookup tables and lookup columns and contains the following topics:

- [Creating Logical Lookup Tables](#)
- [Designating a Logical Table as a Lookup Table](#)
- [About the LOOKUP Function Syntax](#)
- [Creating Logical Lookup Columns](#)

Creating 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 [Designating 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 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.

Designating a Logical Table as a Lookup Table

A logical table must be designated as a lookup table (using the 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 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 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 is 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 is not 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 is 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")`

Creating Logical Lookup Columns

You use the Expression Builder in the 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 is 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 is not 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 is passed in the `WEBLANGUAGE` session variable.

When you use the `DENSE` function (shown in the previous example), if there is 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 cannot use a derived logical column that is 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 is 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 should not have their associated levels below the level of the derived logical column itself.
 - Configuring a descriptor ID column is the recommended approach.

Handling 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

Creating 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 cannot 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 is 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 have 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 have 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 is not 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 repository in the 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 does not 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 is implied by the value column of the LOOKUP function.

Because you cannot 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 is 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 [Creating 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.

Supporting 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 *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for information about Essbase alias tables and how to use them with session variables.

Enabling 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 is not 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 is not selected in the Features tab of the Database dialog in the Administration Tool. See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* 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 are not 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.

Configuring Currency Options

This chapter describes how to configure currency options in Oracle Business Intelligence. 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. This chapter includes the following sections:

- [Changing the Default Currency for Analyses](#)
- [Defining User-Preferred Currency Options](#)

Changing the Default Currency for Analyses

You can change the default currency that is displayed, for example, from French Francs to Euros.

For information about using formatting functions in Answers, see *User's Guide for Oracle Business Intelligence Enterprise Edition*.

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-1" 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-1"` with `tag="int:wrhs"`.

7. Restart the service for Oracle BI Server.

Caution:

The `currencies.xml` file is overwritten when applying an upgrade, 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`.

Use the following procedure to 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 [Defining User-Preferred Currency Options](#).

For more information about INIT BLOCKS, see Working with Initialization Blocks in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Defining User-Preferred Currency Options

You can select the type of currency you prefer to view currency columns in analyses and dashboards.

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

The screenshot shows the 'BI Publisher Preferences' dialog box with the following settings:

- 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 Wizard (limited functionality)
 - Start on Results tab when editing Analysis
- Accessibility Mode: On Off

For information about setting the currency preference on the My Account dialog: Preferences tab or about currency prompts, see *User's Guide for Oracle Business Intelligence Enterprise Edition*.

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 Oracle Business Intelligence repository:

- 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 Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

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.

Defining 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 [Localizing Messages for Users' Preferred Currency](#). Then, for the value of `displayMessage`, specify the `WebMessage` name that is 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 is not 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*

- `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 Business Intelligence.

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.

```
<UserCurrencyPreferences currencyTagMappingType="static">
  <UserCurrencyPreference sessionVarValue="gc1" displayText="Global Currency 1"
currencyTag="int:USD" />
  <UserCurrencyPreference sessionVarValue="gc2" displayText="Global Currency 2"
currencyTag="int:euro-1" />
  <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.



Defining 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*

4. Save and close the `userpref_currencies.xml` file.
5. Restart Oracle Business Intelligence.

Example: Dynamic Mapping to Define User-Preferred Currency Options

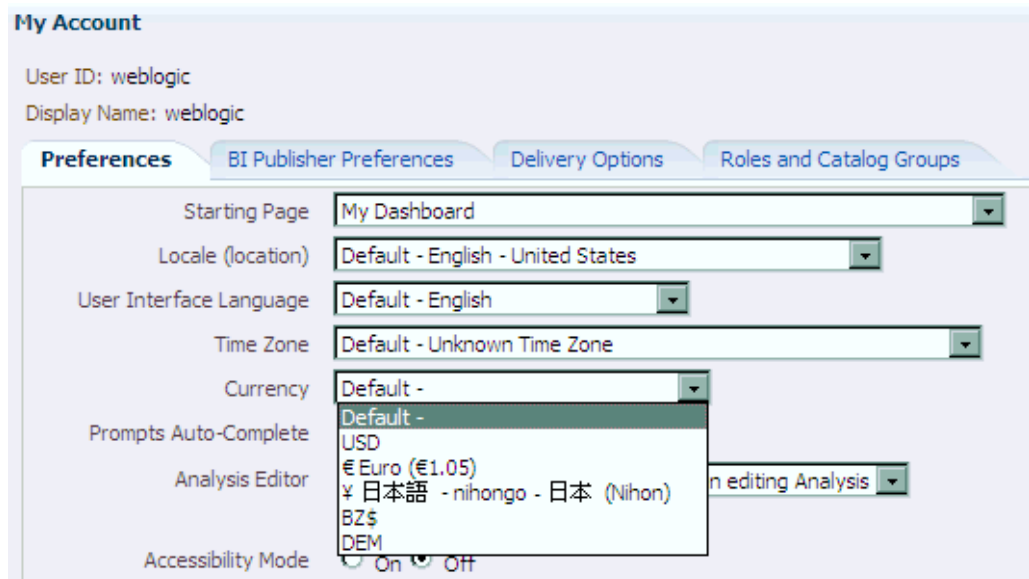
The following example shows a `userpref_currencies.xml` file that uses a dynamic mapping to define user-preferred currency options.

```
<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.



16

Managing How Content is Indexed and Searched

Administrators can set up how catalog and data model content is indexed and crawled so that users find the latest content when they search. By default, the catalog and data models are crawled once a day and all the shared folders are included. You can set up a different schedule to better suit your business and exclude any folders you don't want searched.

Topics

- [Scheduling Regular Content Crawls of Catalog Objects](#)
- [Scheduling Regular Content Crawls of Repository Content](#)
- [Monitoring Search Crawl Jobs](#)

Scheduling Regular Content Crawls of Catalog Objects

It's the administrator's job to select which folders to crawl and schedule when and how often to crawl the content.

1. On the Oracle Business Intelligence Home page, click **Administration**.
2. Under BI Search, click **Configure Crawl**.
3. On the **Catalog** tab of the Configure Crawls page, ensure **Enable Catalog Crawl** is selected.
4. In the **User to Run Crawl As** field, enter an administrative user.
5. For **Languages**, select all the languages for which you want to create indexes.

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 an indexes in both English and Italian.

6. In the **Schedule** section, select the date and time to begin the crawl.
7. Select how often the crawl will be run by providing values in the **Run Every** and **Frequency** fields.

By default the catalog is crawled once daily. When the catalog is updated, the index is updated automatically.

8. To select which catalog objects get indexed, select **Index User Folders** to index the users' private content. By default **Index User Folders** is selected.
9. Select the folders you want the crawl to include by selecting **Index**. Exclude any folders that contain content you don't want others to find when they search by selecting **Don't Index**.
10. Click **Save**.

Scheduling Regular Content Crawls of Repository Content

It's the administrator's job to select which folders to crawl and schedule when and how often to crawl the content.

1. On the Oracle Business Intelligence Home page, click **Administration**.
2. Under BI Search, click **Configure Crawl**.
3. On the **Data Model** tab of the Configure Crawls page, ensure **Enable Data Model Crawl** is selected.
4. In the **User to Run Crawl As** field, enter an administrative user.

The visibility of data and metadata in search results is controlled by the access rights of the administrative user.

5. For **Languages**, select all the languages for which you want to create indexes.

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 an indexes in both English and Italian.

6. In the **Schedule** section, select the date and time to begin the crawl.
7. Select how often the crawl will be run by providing values in the **Run Every** and **Frequency** fields.

By default the data model is crawled once daily. When repositories are updated, you must adjust the next index time to incorporate these updates.

8. Select which subject areas get indexed in the **Select Data Models to Index** section.

By default, all metadata of all subject areas are included in the index. You can select which subject areas, tables, and columns in the subject areas are indexed by expanding the tree of subject areas.

Any element in the hierarchy provides three options:

- **Index Metadata Only:** This is the default selection. This indexes only the metadata associated with the element. For example the column "Product", "Order" or metric names such as "# of Orders".
- **Index:** Use this selection for indexing both the metadata and the data values. This is applicable only to Dimension or Attribute columns. For example, if you select this on "Product" column, then the metadata about Product as well as data values like 'iPad', 'iPod', 'iPhone' are also indexed.
- **Don't Index:** Use this selection to exclude subject areas, tables or columns completely from the index.

9. Click the **Save** icon in the upper right-hand corner.

Monitoring 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 or cancel the next scheduled crawl before it starts.

1. On the Oracle Business Intelligence Home page, click **Administration**.
2. Under BI Search, click **Monitor Crawl**.
The Crawl Job Status page shows information about the past, current, and next scheduled crawl.
3. Look at the **Status** column to find out when the content was last crawled and when the next crawl is due.
You can filter for specific types of Crawl such as Data Model or Web Catalog, and for jobs in a particular status.
4. Click **Cancel** to stop a crawl job that is Running or Scheduled.
5. If a crawl job fails, restart it by returning to the Configure Crawls page and clicking **Enable Data Model Crawl** or **Enable Catalog Crawl** and clicking the **Save** icon.

Configuring and Managing the Oracle BI Presentation Catalog

This chapter describes how to configure and manage the Oracle BI Presentation Catalog and provides information about basic maintenance procedures and configuring for full-text searching.

This chapter includes the following sections:

- [About the Oracle BI Presentation Catalog](#)
- [Maintaining the Oracle BI Presentation Catalog](#)
- [About Catalog Manager](#)
- [Starting Catalog Manager and Opening Catalogs](#)
- [Using the Catalog Manager Workspace](#)
- [Working with Objects in Catalog Manager](#)
- [Validating the Catalog Using Catalog Manager](#)
- [Viewing and Editing Catalog Objects in XML](#)
- [Searching for and Replacing Catalog Text Using Catalog Manager](#)
- [Creating Reports to Display Catalog Data Using Catalog Manager](#)
- [Archiving and Unarchiving Using Catalog Manager](#)

About the Oracle BI Presentation Catalog

The Oracle BI Presentation Catalog stores the content that users create in a directory structure of individual files.

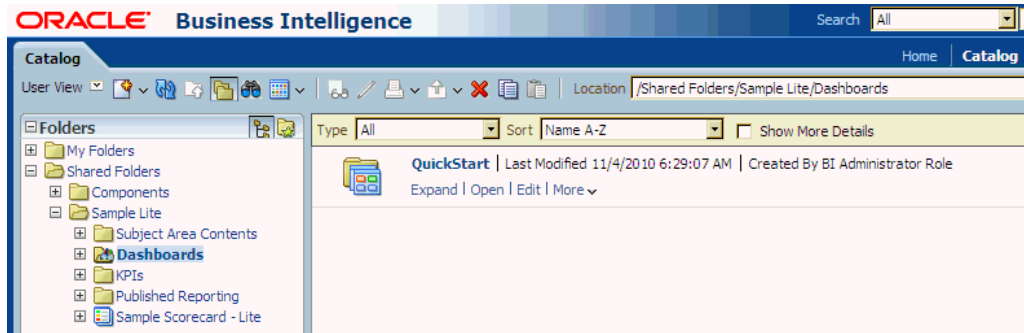
This content includes folders, shortcuts, Oracle BI EE objects (such as analyses, filters, prompts, and dashboards), and Oracle BI Publisher objects (such as reports and templates).

This section contains the following topics:

- [Objects in the Catalog](#)
- [File System Guidelines for Catalogs](#)

Objects in the Catalog

The figure below shows sample objects in the catalog, as seen in Presentation Services.



Guidelines for Object Names

Each object in the catalog is stored in its own file. For example, an analysis called Analysis 1 is stored in a file named Analysis1. 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" \?
```

Attribute Files for Objects

Each object has a corresponding attribute file. For example, the analysis called Analysis1 would have a corresponding attribute file named Analysis1.atr. The attribute file contains the object's full name, access control list (ACL), description, and so on. To access an object in the catalog, users must have appropriate ACL entries for that object. All objects in the catalog use ACL entries.

Lock Files for Objects

To guarantee that only one user can write to a file at one time, a lock file is created when an object is being written to. On rare occasions (for example, after a power outage), temporary lock files in the catalog might not be removed completely. If Presentation Services reports of such a lock file, then you must delete it manually.

File System Guidelines for Catalogs

This section describes the guidelines for working with objects in catalogs in file systems.

Handling Users of the Catalog

The catalog is designed to scale to thousands of concurrent users. To achieve this scaling, the catalog adheres to the following guidelines:

- The average user typically only reads from the catalog and rarely, if ever, writes to it. Each user is constantly and automatically updating his or her Most Recently Used file, but each user's "read" operations still far outweigh the user's "writes" operations. Therefore, the read-to-write ratio is typically at least 100 to 1.
- While a locking mechanism guarantees that only one user can write to an object at a time, it is rare for multiple users to attempt to write simultaneously to the same object. A feature called "lazy locking" enables users to continue reading an object even when another user is updating that object.
- Modern file systems cache "small" files directly inside the directory record, such that reading any information on a directory simultaneously loads all small files directly into the operating system's memory cache. Therefore, it is good practice to keep files in the catalog "small," especially the frequently "read" .atr metadata files. When these metadata files remain small, then all the .atr files in a directory are loaded into memory with one physical hard disk read. Every file that exceeds the "small" threshold adds another physical hard disk read, which can cause a 100% degradation for each large file. In other words, use care when considering storing arbitrary "Properties" in .atr files.

- Reading an object's .atr metadata file using NFS is far slower than reading it directly from a local disk. For this reason, Presentation Services additionally caches all .atr files internally. This cache can become briefly "stale" when another node in the cluster writes data to the file that is newer than the data that is cached by the current node. Therefore, all nodes are refreshed according to the MaxAgeMinutes element in the instanceconfig.xml, whose default for a cluster is 5 minutes. This default setting commonly achieves the best trade-off between the possibility of stale data and the known performance impact. (The default for an environment without clusters is 60 minutes.)

You can modify the MaxAgeMinutes element for your system. Its parent elements are Cache and CatalogAttributes.

Handling Heterogeneous Nodes

To allow heterogeneous nodes in a cluster, the catalog adheres to the following guidelines:

- The maximum length for the name of an object on disk is 256 bytes, which is 64 Unicode characters. The logical name is restricted to 256 Unicode characters. To adhere to this restriction, logical names greater than 32 characters are hashed.
- The maximum length for the name of a path on disk is 32KB, which is 8000 Unicode characters. The logical path is restricted to 16000 Unicode characters.
- All path names on disk are all lowercase. The logical path name allows mixed case, but is still case-insensitive.
- Certain characters are not allowed for path names on disk, while the logical path name allows all characters. For example, Windows systems disallow certain characters such as the colon (:), so those characters are mapped using standard HTML escape sequences. For example, the period character (.) becomes "%2e".
- Certain file names are not allowed on disk, while the logical object name has no restrictions. For example, Windows systems disallow certain file names such as COM, so those names are mapped using standard HTML escape sequences. For example, "com" becomes "co%6d".

Handling Catalog Files on Various Platforms

Keep the following points in mind when handling catalog files on various platforms:

- For UNIX Platforms: UNIX kernels must commonly be configured to allow more than 4000 subdirectories per directory. See [Manually Changing Configuration Settings for the Catalog](#) for information on the HashUserHomeDirectories element.

- For Windows Platforms:

When users want to navigate catalog files using a tool such as Microsoft Windows Explorer, then they want the catalog structure based on a short path name such as c:/obi/demo, rather than the long default path name. Note that such navigation is not recommended.

- FAT is not supported, and NTFS is required.
- Performance on Windows platforms degrades noticeably when more than 8000 files exist in a single directory. Because each catalog object has two files (the data file and the .atr metadata file), it is strongly recommended that you not store more than 4000 catalog objects in a single directory.

- Windows Explorer does not handle long path names properly, and it is recommended to not use Windows Explorer to navigate the internal structure of the catalog. While the file system can handle path names as large as 32KB and Presentation Services is not negatively affected, you cannot use Windows Explorer with any path name that is longer than approximately 2KB.

Because a single Unicode character can require as many as 4 bytes, you might be unable to use Windows Explorer with path names of only 500 Unicode characters. This limitation does not affect Presentation Services. Because of this limitation, place the catalog in a top-level directory, such as c:\mycatalog\sales.

Known Issues with Catalog Files

The following issues are known when working with catalog files:

- Various third-party FTP programs have issues handling '%' escape sequences, which often results in a renamed file that is doubly escaped. For example, a file that is named sa%2epaint (whose logical name is SA.Paint) is incorrectly renamed to sa%252epaint (whose logical name is SA%2ePaint).

Avoid using an FTP program directly against a catalog. Instead, download and use the 7-Zip utility to compress the catalog files, then use an FTP program to transfer the resulting compressed file.

Maintaining the Oracle BI Presentation Catalog

Refer to the topics below for information on maintaining a catalog.

- [Manually Changing Configuration Settings for the Catalog](#)
- [Deploying Catalogs and Objects to Production](#)
- [Updating Catalog Objects](#)

Manually Changing Configuration Settings for the Catalog

You modify settings manually using 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 Catalog section in which you must add the following element:

HashUserHomeDirectories: Specifies the hashing of directories. If you have more than 4000 catalog users or if you intend to have more than 4000 catalog users in the future, then you must turn on the hashing of users' home directories to address a file system limitation. To do so, set the HashUserHomeDirectories element to 2 from its default value of 0. When this element is turned on, for example, the default name for user Steve's home directory would become / users/st/steve.

▲ Caution:

Keep the following points in mind:

- Hashing is not typically needed for modern operating systems that use modern file systems (for example, ext4, ZFS, ntfs); however, it may still be useful for performance when your user base exceeds approximately 4,000 users.
- Oracle recommends that you hash a catalog at creation time; however, there are circumstance that require hashing the catalog after it is in use. Take a backup and run the following command for more information:

On UNIX:

```
runcat.sh -cmd rehash -help
```

On Windows:

```
runcat.cmd -cmd rehash -help
```

- In general, do not set the HashUserHomeDirectories element to a value greater than 2, because a higher setting negates the effectiveness of hashing.
- Include only one Catalog element in the instanceconfig.xml file or unexpected results might occur. Unless expressly noted, include most nodes in an XML document only once.

3. Include the element and its ancestor element as appropriate, as shown in the following example:

```
<ServerInstance>  
<Catalog>  
  <HashUserHomeDirectories>2</HashUserHomeDirectories>  
</Catalog>  
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Presentation Services.

Deploying Catalogs and Objects to Production

You can deploy catalogs and simple objects (for example, a dashboard with privileges) to a production environment from a test environment, as described in the following sections:

- [Deploying Catalogs to Production](#)
- [Deploying Objects to Production](#)

Deploying Catalogs to Production

You deploy a Catalog to production using BAR files.

See [Moving Oracle Business Intelligence Between Environments](#).

Deploying Objects to Production

You can deploy objects (for example, a dashboard with privileges) to a production environment from a test environment.

1. (Optional) If you are deploying a catalog object to a new production environment.

Archive the catalog object in the test environment and unarchive it in the production environment as follows:

- a. Archive the catalog object in the test environment using one of the following:
 - Oracle BI Presentation Services.
 - Catalog Manager.

See [Archiving and Unarchiving Using Catalog Manager](#).

- b. Copy the archived file to the production computer.
- c. On the production computer, unarchive the object.

For information about how to unarchive an object, see [Archiving and Unarchiving Using Catalog Manager](#).

- d. Set the permissions on the object as appropriate.

2. (Optional) If you are deploying the catalog to an existing production environment.

Copy and paste new or updated objects from the test catalog into the production catalog as follows:

- a. Open two Catalog Manager windows: one with the test catalog, and another with the production catalog.
- b. Selectively copy and paste the folders from the test catalog into the production catalog.

If you copy and paste folders where the same content has been changed in the test or production environments, then test content overwrites the production content.

Updating Catalog Objects

If you upgrade to a newer version of Oracle Business Intelligence or install a patch and work with objects in the catalog, then you might notice that certain objects are not being accessed as quickly as in the previous release.

This change can occur if objects are not upgraded 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 upgrade to new versions of Oracle Business Intelligence by following the instructions in the *Upgrade Guide for Oracle Business Intelligence*. To upgrade Catalog objects follow Task 5: Upgrade the Oracle BI Repository and Catalog in *Upgrade Guide for Oracle Business Intelligence*. Oracle recommends that you upgrade when Presentation Services is not running. If you suspect that the upgrading of objects was not performed thoroughly during the upgrade process, 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 are performing a rolling upgrade of machines in a cluster, then do not use this option or the UpgradeAndExit configuration setting until all machines in the cluster are upgraded.
 - 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 were not. You can view the log files for details on objects that were not updated.

About Catalog Manager

Catalog Manager is a tool that lets you perform online and offline management of Oracle BI Presentation Catalogs. Install Catalog Manager on a secure computer that is accessible only to Oracle BI Administrators.

Uses for Catalog Manager

You can use Catalog Manager to:

- Manage folders, shortcuts, global variables, and objects (analyses, filters, prompts, dashboards, and so on). For example, you can rename and delete objects, and you can move and copy objects within and between catalogs.
- View and edit catalog objects in Extensible Markup Language (XML).
- Preview objects, such as analyses and prompts.
- Search for and replace catalog text.
- Search for catalog objects.
- Create analyses to display catalog data.
- Validate the catalog.
- Localize captions. See [Localizing Oracle BI Presentation Catalog Captions](#).

Many of the operations that you can perform in Catalog Manager can also be performed through the Catalog page in Oracle BI Presentation Services. See Managing Objects in the Oracle BI Presentation Catalog in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

Guidelines for Working with Catalog Manager

Follow these guidelines when working with Catalog Manager:

- Always make backup copies of the Oracle BI Presentation Catalogs that you are working with.
- Be sure of changes that you plan to make. Catalog Manager commits changes immediately. There is no undo function nor are there any error messages to tell

you that a particular change does not display well to users. However, if you do make any unwanted changes, then you can revert to your latest saved backup.

- Do not copy and paste catalog contents into email, as this is not supported.

Tips for Working with Catalog Manager

As you work with Catalog Manager, keep the following tips in mind:

- While working in online mode, you can paste catalog contents into or out of a read-only folder by turning off the read-only property of the folder tree before copying, then re-apply the read-only attribute after pasting.
- You cannot copy, archive, or drag files from the `/system/security` directory in the Catalog Manager.
- Some keyboard shortcuts might not work properly.
- Even if a resize indicator is not shown, Catalog Manager panes might still be resizable.
- You can use Catalog Manager in languages other than English. See [Setting the Current Locale in Catalog Manager](#).

Starting Catalog Manager and Opening Catalogs

Learn more about starting catalog manager and opening catalogs in these topics.

This section describes the following topics:

- [Requirements for Running Catalog Manager](#)
- [Starting the Catalog Manager User Interface](#)
- [Resolving Startup Issues on Linux Systems](#)
- [Understanding the Two Catalog Modes](#)
- [Operations Available in Online Mode and Offline Mode](#)
- [Opening an Oracle BI Presentation Catalog](#)

Requirements for Running Catalog Manager

You must use these components to run Catalog Manager.

The following list describes the requirements for running Catalog Manager:

- **Graphical User Interface** — You can invoke the user interface on the following platforms:
 - Windows 64-bit
 - Linux 64-bit
- **Command Line Utility** — You can invoke the command line utility on supported platforms for Oracle Business Intelligence such as Windows, Linux, IBM-AIX, Sun Solaris, and HP-UX. Enter a command such as the following one on Linux for assistance in using the command line utility:

```
./runcat.sh -help
```

Starting the Catalog Manager User Interface

You can start the user interface for Catalog Manager using menu options on Windows or a command on Windows or Linux.

1. On Windows, from the Start menu, select **Oracle Business Intelligence**, then **Catalog Manager**.

or

Using the command line, change to the following directory:

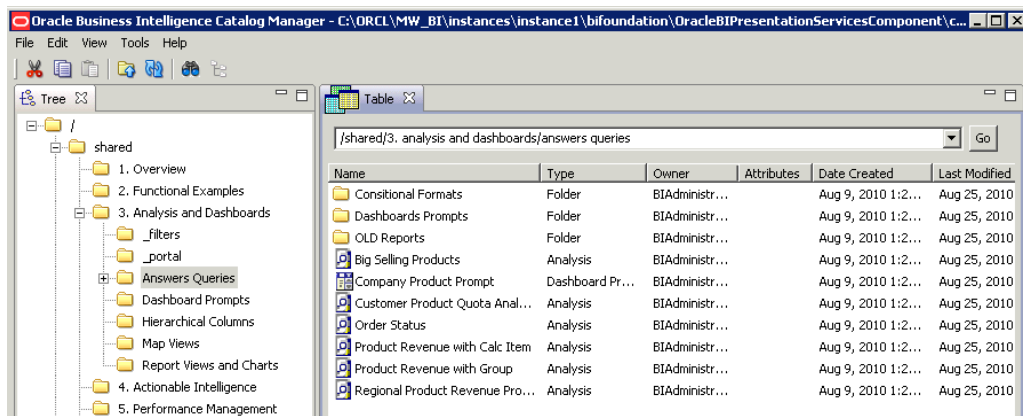
```
BI_DOMAIN\bitools\bin
```

then run the appropriate script:

```
runcat.cmd (on Windows)
```

```
runcat.sh (on Linux)
```

The following illustration shows sample objects in the Catalog Manager for Windows platforms.



Resolving Startup Issues on Linux Systems

You must start the Catalog Manager in a graphical user interface xterm on Linux systems.

Examples of xterms are a native gnome, kde console, VNC, or a local x-server such as Xming, Tarantella, Hummingbird Exceed, or Citrix. (These examples do not constitute a statement of certification or support.) You cannot start the graphical user interface for Catalog Manager using an ASCII text terminal, such as PuTTY or FSecure or a command-line SSH.

If the Catalog Manager does not start, then verify the following:

- That you can run a native application such as xclock or xeyes.
- That you can start Catalog Manager with a native console or with VNC. Sometimes operating system administrators can lock X-Windows.
- That you can run the following command, which allows all xterm connections:

```
xhost +
```

- That you can run Catalog Manager from the command line with debugging enabled to see if any additional output is produced, using the following command:

```
./runcat.sh -consoleLog -noExit
```

- That you can use an operating system utility such as `strace` to trace the execution of the `runcat.sh` command and see if any error messages are generated, such as those relating to libraries or files being unable to open. You can use the Eclipse Java plug-in that requires the standard widget toolkit (SWT), which in turn requires GTK (Gimp Toolkit) to be installed. Enter the following command:

```
strace -aef -o ./runcat_trace.txt runcat.sh
```

Understanding the Two Catalog Modes

You can open a catalog in one of two modes — online or offline. Both modes can operate against an actual production catalog, with no need for any downtime.

Online Mode

In online mode, you connect to a catalog on a running web server. In this mode your permissions are applied, you can select a locale, and you can see the effects of any localization on the catalog. You can see only those objects for which you have the appropriate permissions. Both Presentation Services and the web server must be running for you to open catalogs in online mode.

Use online mode when you want to make minor incremental changes or additions to the catalog, such as changes to permissions, updates to a single object, or migration of new objects to a production environment.

Offline Mode

In offline mode, you connect to a local file system. In this mode, you are logged in as a super user or system user, and no permissions are applied. You can see all objects in the catalog.

Generally, working in offline mode is faster than working in online mode. This is because you are accessing, creating, and updating the individual files directly, and the catalog does not have to communicate with Presentation Services as it does when you are working in online mode.

Use offline mode when you want to make catalog-wide changes, such as globally renaming objects or moving multiple objects for reorganization, as described in the following procedure.

1. Place Presentation Services in Maintenance Mode.

In a clustered environment, you can place any instance of Presentation Services in Maintenance Mode and within a few minutes, all other instances in the cluster automatically go into Maintenance Mode too. In a clustered environment only, wait a few minutes before performing the tasks for which you placed Presentation Services into Maintenance Mode.

2. Back up the catalog by using the 7-Zip utility to create a compressed file for it.
3. In Catalog Manager, open the catalog in offline mode, as described in [Opening an Oracle BI Presentation Catalog](#).
4. Make the systemwide change.
5. Back up the catalog again.

6. Take Presentation Services out of Maintenance Mode.

In a clustered environment, you can take any instance of Presentation Services out of Maintenance Mode and within a few minutes, all other instances in the cluster automatically go out of Maintenance Mode too. So in a clustered environment only, a few minutes are required before users can resume editing catalog content.

Presentation Services Maintenance Mode

Maintenance Mode is set on the Presentation Services Administration page. When you put Presentation Services into Maintenance Mode, you make the catalog read-only so that users cannot modify its contents. Users can continue to perform tasks that do not involve writing to the catalog such as viewing existing objects (for example, dashboards and analyses). Tasks such as updating a user's "most recently used" list are not available.

When Presentation Services is in Maintenance Mode, a lock file that is named `root_maintenancemode.lock` is created in the directory for the catalog. (On UNIX systems, that file is a link to a file that has a name such as `root_maintenancemode_number.lock`, meaning both files make up the lock file.) If you create a zip file for the catalog when Presentation Services is in Maintenance Mode, then the zip file includes the lock file. If you extract the contents of the zip file and open the catalog, then Presentation Services is placed in Maintenance Mode.

Unlike previous releases, if you put Presentation Services into Maintenance Mode and Presentation Services is restarted, then Presentation Services starts with Maintenance Mode still turned on.

Operations Available in Online Mode and Offline Mode

Many of the operations that you can perform using Catalog Manager are available in both online mode and offline mode. A few operations are available in only one mode or the other.

Generally, the operations available in:

- Online mode includes read-only operations and write operations that do not affect the entire catalog, such as setting permissions for an object.
- Offline mode includes most of the operations available in online mode and write functions that affect the entire catalog.

You can perform the following operations in online and offline modes (or as stated), as follows:

- 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. (This feature is known as Smart Rename and is available in both modes. In offline mode, you can rename all objects. In online mode, you might be unable to rename certain objects, depending on your permissions.)
- Refreshing the Catalog Manager workspace.
- Creating folders.
- Setting permissions for objects.
- Working with properties of objects.
- Managing the view of the workspace.
- Searching for objects.
- Searching for and replacing catalog text. (This feature is available in both modes. In offline mode, you can replace all objects. In online mode, you might be unable to replace certain objects, depending on your permissions.)
- Creating reports to display Catalog Manager data.
- Setting browser preference.
- Previewing objects (available in online mode only).
- Exporting captions for localization purposes.

Opening an Oracle BI Presentation Catalog

Follow the steps below to open an Oracle BI Presentation Catalog.

1. In Catalog Manager, from the **File** menu, select **Open Catalog**.
2. Complete the necessary fields, as described in the following list:
 - **Type** — Select the mode (online or offline) in which to open the catalog
 - **Path** — If you are opening the catalog in offline mode, then enter the path to the catalog folder on the local file system, for example:
Click **Browse** to display a dialog for locating the catalog.
`SDD/metadata/content`
where SDD is the Singleton Data Directory for example, `DOMAIN_HOME/bidata`. See [Key Directories in Oracle Business Intelligence](#).
 - **URL** — If you are opening the catalog in online mode, then enter the URL to Oracle BI Presentation Services, for example:
`https://hostname/analytics/saw.dll`
 - **User** — If you are opening the catalog in online mode, then enter the user name for the host URL (disabled in offline mode).
 - **Password** — If you are opening the catalog in online mode, then enter the password for the host URL (disabled in offline mode).
 - **Locale** — Select the locale to use for user interface elements in Catalog Manager and for objects in the catalog, when opening the catalog in online mode.
 - **Read-Only** — Select this field to open the catalog in read-only mode (disabled in offline mode).

3. Click **OK**.

When specifying the URL for the catalog in online mode, ensure that you specify `https` rather than `http`, for increased security. If you specify `http`, then you see a message box after closing the Open Catalog dialog that prompts you to verify the opening of the catalog using an unsecured connection. To use `https` when opening catalogs, you must configure Oracle BI EE for Secure Socket Layer communication, as described in *Configuring SSL in Oracle Business Intelligence in Security Guide for Oracle Business Intelligence Enterprise Edition*.

Using the Catalog Manager Workspace

The Catalog Manager workspace provides resources for maintaining user content.

This section provides the following topics on the workspace for Catalog Manager:

- [What Does the Catalog Manager Workspace Do?](#)
- [What Does the Catalog Manager Workspace Look Like?](#)
- [Managing the View of the Catalog Manager Workspace](#)

What Does the Catalog Manager Workspace Do?

The Catalog Manager workspace enables you to view and work with catalog objects.

It displays the following folders for an open catalog:

- The shared folder — Contains content that is shared among catalog users. This includes the preconfigured dashboards and analyses that are distributed with prebuilt applications, and other objects such as shared filters.
- The system folder — Contains administrative elements of Presentation Services. Some of these elements are distributed with the product, and others are configured by you as the administrator, such as privileges. Avoid modifying any files in this folder. Presentation Services uses these files internally and modifying them might cause unexpected results.
- The users folder — Contains content that catalog users with the appropriate permissions have saved to their personal folders, such as individual analyses.

What Does the Catalog Manager Workspace Look Like?

The Catalog Manager workspace provides a variety of tools for working with data.

Catalog Manager consists of the following main components:

- Menu bar — Lets you access the following menus:
 - File — Provides options that let you open and close catalogs, exit Catalog Manager, and so on.
 - Edit — Provides options that let you manage catalog objects, such as Cut, Copy, Permissions, and so on. (Many of these options are also available on the right-mouse menu.)
 - View — Provides options to manage the view of the Catalog Manager workspace, such as **Show Tree**, **Show Job Status**, and so on.

- Tools — Provides options that let you manage catalogs, such as **XML Search and Replace**, **Create Report**, and so on.
- Help — Provides options to access the Oracle BI Enterprise Edition website and to display information about Catalog Manager.
- Toolbar — Provides quick access to commonly used options, such as Cut, Copy, Paste, and so on.
- Tree pane — Displays catalog folders. The pane also displays objects but only if the **Show Objects in Tree** option on the View menu is selected.
- Table pane — Displays catalog folders and objects. It consists of:
 - The navigation bar, where you can move to the catalog object to work with by typing its path name.
 - These columns: Name, Type, Owner, My Permissions, Attributes, Date Created, and Last Modified. Click the column name to sort by that value, such as by type.

The Type column identifies the type of object. Objects that are identified as "unknown file" are generally internally used objects, and their type is not exposed in Catalog Manager.
- Right-mouse menu — Provides options that let you manage catalog objects, such as Rename, Properties, Permissions, and so on. (Many of these options are also available on the Edit menu.)

Managing the View of the Catalog Manager Workspace

You can manage what you view in the Catalog Manager. For example, you can show objects in the Tree pane or show job statuses.

1. In Catalog Manager, select **View** and then one of these options:
 - **Show Tree** — Displays the Tree pane, if you previously had closed it.
 - **Show Table** — Displays the Table pane, if you previously had closed it.
 - **Show Job Status** — Displays the Background Job Status pane, where you can view the progress of processes that you have run, such as Search and Replace, Smart Rename, and so on. You can also remove all finished jobs and set progress preferences using the icons in the upper-right corner of the pane.
 - **Show Objects in Tree** — Displays objects (that is, analyses, filters, and so on) in addition to folders in the Tree pane.
 - **Refresh** — Refreshes the objects that are displayed in the Tree and Table panes. (You might want to refresh the data, for example, if someone else makes changes to the catalog while you are working with it and you want to see the changes.)

Working with Objects in Catalog Manager

You can alter objects in Catalog Manager in several ways.

This section provides the following information about working with objects:

- [Searching for Catalog Objects Using Catalog Manager](#)

- [Copying and Pasting Objects](#)
- [Renaming Catalog Objects](#)
- [Working with the Properties of Catalog Objects](#)
- [Setting Permissions of Catalog Objects](#)
- [Previewing Objects from Catalog Manager](#)

In the Catalog page of Presentation Services, you can view folders and contents including hidden objects. You can create, rename, copy, move, and delete folders and contents. See *Managing Objects in the Oracle BI Presentation Catalog* in *User's Guide for Oracle Business Intelligence Enterprise Edition*.



Note:

Changes made in the Presentation layer of the Oracle BI Administration Tool can affect analyses and dashboards based on those tables and columns. You can use Catalog Manager to keep the catalog synchronized with these changes in the Presentation layer.

Searching for Catalog Objects Using Catalog Manager

You can search for objects in the catalog using the Search function.

For example, you might want to search for all objects that have a property with the value of "administrator."

When you search, you can limit the search by:

- **Case Sensitive** — Select this check box to apply case sensitivity to the search criteria. The default value is unchecked.
- **Name** — Limits the search to the names of objects.
- **Description** — Limits the search to the Description property.
- **Property values** — Limits the search to the values of properties.
- **Owner** — Limits the search to the owners of objects.
- **XML** — Limits the search to XML.
- **Object type** — Searches for all types of objects or limits the search to a specific type of object that you specify (for example, analyses, filters, agents, dashboard prompts, dashboard pages).
- **Date** — Limits the search to objects that were created on the dates that you specify or to objects that were last modified on the dates that you specify.

Use the following steps to search for an object.

1. In Catalog Manager, open the catalog and navigate to the location in the tree where you want to begin the search.
2. Click **Search** on the toolbar.
3. In the **Search for any or all criteria below** field, enter the word or phrase to search for.
4. To make the search case-sensitive, select the **Case Sensitive** box.

5. To limit the search, then click **Advanced Search**.
6. In the Advanced Search area, specify the constraints for the search.
7. Click **Search**.

 **Tip:**

When you have finished searching, click **Explore the entire catalog tree** on the toolbar to return to the Tree and Table panes.

Copying and Pasting Objects

You can copy and paste objects within a single catalog.

You can also copy objects from one catalog and paste them into another catalog.

Tips for Copying and Pasting

You can execute and use copying and pasting in several ways.

Use the following tips as you copy and paste objects:

- You can copy and paste objects using the following methods:
 - Menu options, as described in [Copying and Pasting Objects Between Catalogs Using Menus](#).
 - Drag and drop, to copy objects between two catalogs and within the same catalog. Drag and drop always makes a copy of the dragged objects, even when performing a drag and drop within a single catalog.
 - Archive and unarchive, as described in [Archiving and Unarchiving Using Catalog Manager](#). When you archive, you create a file that you can save for later use. The unarchiving process automatically overwrites any files without providing the opportunity to specify that certain files not be overwritten.
- Catalogs are structured in hierarchical folders. When copying or merging objects, remember to also copy any objects that are associated with them, such as dashboard folders, shortcuts, and analyses. URL paths in external applications can be reestablished after a copy or merge operation if the entire folder path is not copied, for example, if added to the dashboard as a shortcut or text.
- Most often, you can simply copy and paste objects as needed. If required, you can set advanced options that affect the objects that you are pasting. See [Advanced Options for Pasting Objects](#).

Copying and Pasting Objects Between Catalogs Using Menus

The following procedure describes how to copy and paste objects between two catalogs using menu options.

If the two catalogs have the same name, then you might want to rename one of the catalogs before opening it to help distinguish between the two catalogs as you work. Both catalogs must be the same version 11.1.1 (or later).

1. In Catalog Manager, open the catalog that is to be changed (the target catalog).

2. Using another instance of the Catalog Manager, open the catalog that contains the objects to copy from (the source catalog).
3. If necessary, reposition both instances of Catalog Manager on your screen so you can display the title bars of both Catalog Manager instances.
4. In the Catalog Manager instance that shows the source catalog, right-click the source object and select **Copy**.
5. In the Catalog Manager instance that shows the target catalog, right-click at the point where you want to paste the source object and select **Paste**.

Advanced Options for Pasting Objects

You can set advanced options in the Preferences dialog for pasting objects that you have copied, as described in the following sections:

- [Paste Overwrite](#)
- [Paste ACL](#)

Caution:

You must set the advanced options in the Preferences dialog before you begin the copy and paste operation, for them to take effect.

Paste Overwrite

The Preferences dialog contains a number of options in the Paste Overwrite area.

Options include:

- **Force** — Pastes all files, overwriting even those that have the read-only attribute set.
- **All** — Pastes all possible files, overwriting only those that do not have the read-only attribute set. (Default)
- **Old** — Pastes all possible files, but does not overwrite any existing files unless they are older than the source.
- **None** — Pastes all possible files, but does not overwrite any existing files.

Consider the following example of pasting with overwrite options set. Suppose that the /users/joe folder contains the following analyses:

Analysis A (created 01-Jan-2010)
Analysis B (created 31-May-2010)
Analysis C (created 01-Jan-2010)

Suppose that the /users/sue folder contains the following analyses, but no Analysis C

Analysis A (created 28-Feb-2010)
Analysis B (created 01-Jan-2010)

Suppose that Sue copies the A, B, and C Analyses from the /users/joe folder and pastes them to the /users/sue folder. If the Paste Overwrite option is set to:

- **None**, then Sue keeps her A and B Analyses, and Joe's analyses are ignored. Sue gets a copy of Analysis C.
- **All**, then Sue's A and B Analyses are overwritten with Joe's, and Sue gets a copy of Analysis C.
- **Old**, then Sue keeps her A Analysis (Sue's A Analyses is not "old"), Sue's B Analysis gets overwritten by Joe's analysis (Sue's B Analysis was "old"), and Sue gets a copy of Analysis C.

Paste ACL

The Preferences dialog contains a number of options in the Paste ACL area.

Options include:

- **Inherit** — Inherits the object's permissions (ACL) from its new parent folder. (Default)
- **Preserve** — Preserves the object's permissions (ACL) as it was in the original, mapping accounts as necessary.
- **Preserve Only Groups** — Same as **Preserve**, but applies to group accounts and Application Roles, not to user accounts. This is for a development to production environment in which a customer might use the same groups (such as Sales and Marketing) in both development and production. However, the users in each group might be very different, such as TestUserA and TestAdminB in development and Steve and Sue in production.
- **Create** — Preserves the object's permissions (ACL) as it was in the original, creating and mapping accounts as necessary, depending on the mode and type of owner, as described in the following list:
 - **Online mode** — In online mode, Catalog Manager is communicating with the back-end security server. Catalog Manager knows about the users and application roles from that server and can usually paste a copied object with the appropriate user name or role. While pasting objects, keep in mind that you might lack appropriate permissions to create accounts for certain objects.
 - **Offline mode** — In offline mode, Catalog Manager has no connection with the back-end security server, so it is unaware of users and application roles that are stored there, unless their names are available in the cache for the catalog. If the name of the user or role for a copied object is not available in the cache, then Catalog Manager cannot paste the copied object with that name or role. Instead, the pasted object inherits its owner from its new parent folder, which is similar to the **Inherit** option.

This feature is used in applications whose administrators create accounts in a staging area before moving the users to the production environment.

If you have the appropriate permissions, then you can select a newly pasted object and set ownership recursively to the appropriate user.

Consider the following example of pasting with ACL options set. Suppose that Steve owns the /users/steve/MyFavReport folder and has permissions (ACL) "all users can read/execute, steve has full control". Joe (who has some administration privileges) logs in and copies MyFavReport, pasting it to /users/sue (which is owned by "administrator", with permissions "admins have full control, sue has full control").

If Joe sets the Paste ACL option to:

- **Inherit**, then the `/users/sue/MyFavReport` folder is owned by Joe with whatever permissions are set on the `/users/sue` folder (that is, "admins have full control, sue has full control").
- **Preserve**, then the `/users/sue/MyFavReport` folder is owned by Joe with whatever permissions were set on the `/users/steve/MyFavReport` folder (that is, "all users can read/execute, steve has full control"). If Joe pastes in a second Catalog Manager and if "steve" does not exist in this Catalog, then the permissions for Steve are discarded. If "steve" exists but has a different user ID, then Steve's user ID is mapped to the new one.
- **Create** in online mode, then the `/users/sue/MyFavReport` folder is owned by Joe with whatever permissions were set on the `/users/steve/MyFavReport` folder (that is, "all users can read/execute, Steve has full control"). If Joe pastes in a second Catalog Manager and if "steve" does not exist in this catalog, then the owner is inherited from the parent folder. (The Create option is deprecated in Release 11g as it applies only to Catalog groups.)

Renaming Catalog Objects

You can rename objects in the catalog.

This can be useful when you are migrating from a test environment to a production environment.

There are two ways to rename an object:

- **Rename without reference updates** — Renames the object and preserves the references to the original name that other catalog objects might have.
- **Rename with reference updates** — Renames the object and changes references that other objects might have to the new name (that is, original name references are not preserved). This feature is also known as Smart Rename. You can open the catalog in either offline or online mode. In offline mode, you can rename all objects. In online mode, you might be unable to rename certain objects, depending on your permissions.

Caution:

Keep the following points in mind when renaming objects:

- You cannot rename a user account in the catalog. If you rename a user's home directory, then you do not rename that user and you might see unexpected results.
- The catalog contains several reserved names of objects. Rename only your own objects, not those that Presentation Services creates internally. For example, do not rename the `_portal` directory in your home directory, because then you cannot see "My Dashboard".

Use the following steps to rename an object without reference updates.

1. In Catalog Manager, open the catalog.
2. Navigate to the object to be renamed.

3. Right-click the object in the Name column and select **Rename**.
4. Type a new name for the object.

To rename an object with reference updates:

1. In Catalog Manager, open the catalog in offline mode.
2. Navigate to the object to be renamed.
3. Right-click the object in the Name column and select **Smart Rename**.
4. Type a new name for the object.

A progress bar in the lower-right corner of the window shows the progress of the reference updates.

Working with the Properties of Catalog Objects

You can work with object properties through the Catalog Manager.

Using the Properties option of Catalog Manager, you can:

- Create, view, edit, and delete the properties of catalog objects.
 - Change attributes of catalog objects to hide them from display in Oracle Business Intelligence.
1. In Catalog Manager, open the catalog.
 2. Navigate to the object.
 3. Right-click the object in the Name column and select **Properties**.
 4. Perform the necessary tasks:
 - a. If you have the appropriate permissions, then select the appropriate owner for the object in the **Owner** list.

The Owner list includes the name that you used to log in to Catalog Manager. You can use this list to select yourself as the owner of the object.

See *Assigning Ownership of Objects in User's Guide for Oracle Business Intelligence Enterprise Edition* for information on taking ownership of objects.

- b. To change the attribute of an object, select either **Read-Only** or **Hidden**, if appropriate. A hidden object is not visible in Oracle Business Intelligence.

Note:

The **System** option indicates that the object is maintained internally and should not be altered.

- c. To create, edit, or delete a property, use the **New**, **Edit**, or **Delete** button as appropriate.

 **Note:**

The **New** button is used to create a property. Use it only if instructed to do so by Oracle Support Services.

5. Click **OK**.

You can select multiple objects and update their properties or permissions simultaneously. If any of the selected objects are a folder, then you can also apply those changes recursively to all the objects in that folder's tree structure.

For example, you can set all files in the /shared/DontTouch directory to be Read-Only. Right-click the DontTouch directory and select **Properties**. In the Properties dialog, select the **Read-Only** option, select the **Apply Recursively** option, and click **OK**. You can also select **Apply Recursively** to take ownership of an object and all its sub-objects.

Setting Permissions of Catalog Objects

Permissions are used to control access to catalog objects.

1. In Catalog Manager, open the catalog.
2. Navigate to the object.
3. Right-click the object in the **Name** column and select **Permissions**.

The Permissions dialog displays these two lists:

- **Users and Application Roles (Explicit Permissions)** — Shows the users, groups, and application roles that have explicit permissions granted to this object.
- **Additional Users and Application Roles** — Shows the users, groups, and application roles that have access granted through group inheritance, and users, groups, and application roles that have no access to the request.

For details on how permissions and privileges are assigned in Presentation Services, see Managing Security for Dashboards and Analyses in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

4. If the user, group, or application role whose permissions you want to set is in the **Additional Users and Application Roles** list, then move it into the **Users and Application Roles (Explicit Permissions)** list by selecting it and clicking the left-arrow button (<).
5. (Optional) To filter the users, groups, and application roles displayed in the **Additional Users and Application Roles** list, use the **List** button and the adjacent field, as follows:
 - Enter filter criteria in the field next to the **List** button (case insensitive) to search by name.
To enter partial filter criteria, use the asterisk (*) symbol. For example, enter `bi*` to display users or groups beginning with `bi`, `BI`, `bl`, and `Bi`.
 - Select a value from the list, to restrict what accounts to search for.
Available values are: All, User, or Application Role.

6. Select the user or group in the **Users and Application Roles (Explicit Permissions)** list.
7. Select a new permission from the list in the Permissions column, or click **Custom** from the list to display the Custom Permissions dialog, where you can select a combination of permissions.

For details on what each permission means, see Permission Definitions in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

8. Select the **Apply Recursively** option to apply the changes to all the objects that the object contains.
9. Select a value from the **Replace Option** list as follows:
 - **Replace All** — Replaces the existing ACL with what is currently in the dialog.
 - **Replace Listed** — Changes only the accounts currently displayed in the dialog and leaves others unchanged.
 - **Remove Listed** — Removes only the accounts currently displayed and leaves others unchanged.
10. Click **OK**.

 **Note:**

If you move a user or group from the **Users and groups (Explicit Permissions)** list to the **Additional Users and Application Roles** list, then the user or group privileges are reset to No Access. To move a user or group from one list to another, highlight it and click the right or left-arrow button, as appropriate.

See *Security Guide for Oracle Business Intelligence Enterprise Edition* for more information about permissions and groups and users.

Previewing Objects from Catalog Manager

You can preview objects, such as analyses or prompts, from Catalog Manager in online mode. If you are going to preview objects from Catalog Manager, then you must identify the default browser in which to display these objects.

1. In Catalog Manager, from the **Tools** menu, select **Preferences**.
2. In the **Select Web Browser to use for report previews** field, select the browser that is the same one that you have set to be the default browser for your operating system. You can click the **Browse** button in which you can select the executable file for the appropriate browser.
3. Click **OK**.

To preview an object:

1. In Catalog Manager, open the catalog in online mode.
2. Navigate to the object.
3. Right-click the object in the Name list and select **Preview**.

Validating the Catalog Using Catalog Manager

The catalog is a dynamic component which requires maintenance. Over time, inconsistencies can develop in the catalog as links are broken, users are deleted, or NFS file system issues are encountered. These inconsistencies can eventually lead to incorrect behavior, such as the inability to edit an agent's recipient list. You can periodically take the production system offline and validate the catalog, to be informed of and to take corrective action on inconsistencies.

The validation process performs the following tasks:

- Ensures that each object in the catalog is larger than zero bytes.
- Ensures that each item in the catalog has a valid corresponding `.ATR` file.
- Ensures that each link in the catalog is valid.
- Ensures that all XML objects in the catalog pass schema validation.
- Attempts to repair object names that were damaged by ftp programs.

Prerequisites

Before validating the catalog in a clustered environment, do one of the following:

- Shut down the Presentation Services cluster and run the validation directly against the cluster's catalog.
- Make a copy of the cluster's catalog and run the validation against that copy.

Before using the 7-Zip utility to make a copy of a catalog, first shut down all nodes in the Presentation Services cluster or put all nodes in that cluster into Maintenance Mode (which is the recommended approach).

Be aware that any changes that are made to the catalog online concurrently to the validation process are not included in the validation.

While backing up the catalog is always good practice, there is no practical difference between running `validate` against the catalog directly versus running the validation on a backup copy.

Validating the Catalog

Catalog validation can only be completed using the command line, not the Catalog Manager user interface.

1. Open the catalog in offline mode as described in [Understanding the Two Catalog Modes](#).
2. The basic syntax to validate the catalog is:
 - Windows: `runcat.cmd -cmd validate`
 - UNIX: `runcat.sh -cmd validate`

For a list of parameters used by the command and information on how to use them, append `-help` to the command.

Viewing and Editing Catalog Objects in XML

Catalog Manager provides the ability to view and to edit the XML description of catalog objects such as analyses, dashboards, filters, and so on.

While viewing the XML code is acceptable, editing the code is not recommended.

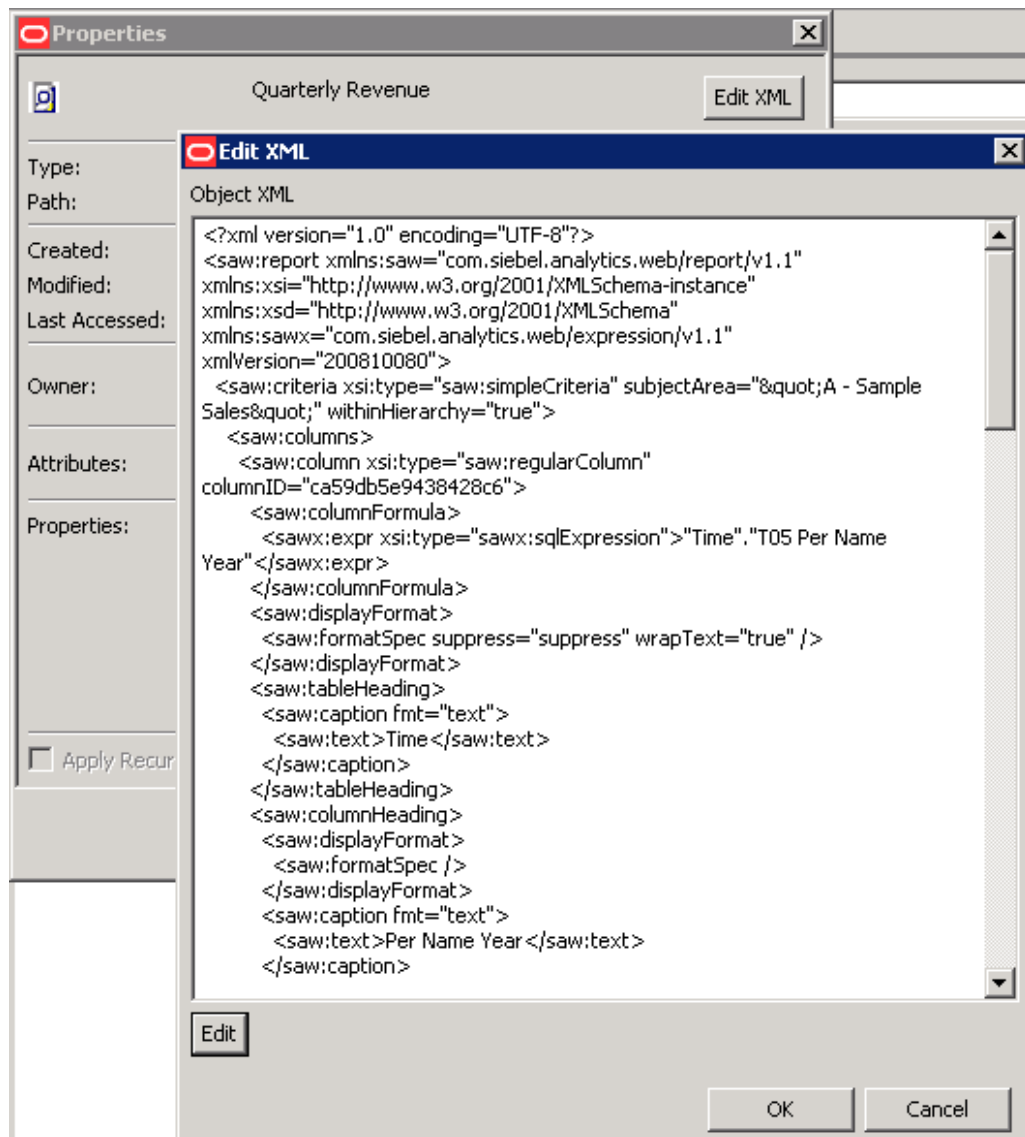
▲ **Caution:**

If you edit the XML code, then you change the representation of the object in the catalog. Editing the XML code for catalog objects in any directory is not recommended and can produce unexpected results.

If you want to edit the XML for an analysis, then use the information in *Examining the Logical SQL Statements for Analyses* in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

1. In Catalog Manager, open the catalog.
2. Navigate to the object.
3. Right-click the object in the Name column and select **Properties**.
4. Click **Edit XML**.
5. When you have finished viewing the XML definition, click **Cancel**.
6. Click **OK** in the Properties dialog.

The illustration shows sample XML code in Catalog Manager for an object.



To edit the XML description of an object, which is not recommended:

1. In Catalog Manager, open the catalog.
2. Navigate to the object.
3. Right-click the object in the Name column and select **Properties**.
4. Click **Edit XML**, then **Edit**.
5. Make the changes in the Object XML area.

 **Note:**

When you edit the XML description of an object, the catalog checks only that the XML is well-formed; it does not check for any other errors.

6. Click **OK** in the Edit XML dialog.

7. Click **OK** in the Properties dialog.

Searching for and Replacing Catalog Text Using Catalog Manager

You can search for specific text in the catalog and replace it with other text using Catalog Manager.

You can open the catalog in either online or offline mode. In offline mode, you can replace all objects. In online mode, you might be unable to replace certain objects, depending on your permissions.

Specifically, you can search for and replace:

- A simple text string using a dialog, as described in [Searching for and Replacing a Simple Catalog Text String](#).

For example, suppose that an object contains the string "My Misspeled Wirlds." You can use Catalog Manager to search and replace that string with the proper text of "My Misspelled Words."

- Multiple or complex text strings all at the same time using an XML file, as described in [Searching for and Replacing Multiple Catalog Text Strings](#).

For example, suppose that the administrator renames a subject area, a table, or column in the repository file. The table "Sales" might be renamed "MySales." You can use Catalog Manager to search and replace all uses of that object throughout the catalog.

Searching for and Replacing a Simple Catalog Text String

You can search for a simple text string in the catalog and replace it with other text.

1. In Catalog Manager, open the catalog in either online or offline mode.
2. From the **Tools** menu, select **XML Search and Replace**.
3. In the **Old text** field, enter the text string to search for.
4. In the **Replace with** field, enter the replacement text.
5. To make the search case insensitive, deselect the **Case Sensitive** box.
6. Click **OK**.

About Searching for and Replacing Multiple Catalog Text Strings

You can perform more powerful search and replace operations on multiple catalog text strings all at the same time by importing a XML file that identifies each text string to search for and replace.

XML File Format for Searching for and Replacing Text Strings

In the search and replace XML file, you use an action element to identify each text string to search for and replace.

The action elements are contained in a commands element. The action element has the following attributes:

- **command** — Specifies the text to replace. The valid value is:
 - **textReplace** — Replaces all the text that matches in an XML file, such as a column name.
- **oldValue** — Specifies the text string to search for.

When you specify this attribute for the `textReplace` command for the search and replace XML file, you must use the full Java regex syntax, which is not like a normal string. To replace a string, you must do the following:

1. Escape any special Java regex characters (such as brackets, parentheses, dollar signs, and carets).
2. Escape any special "normal" string characters (such as back slashes and quotes).
3. Because you are working in an XML file, escape any special HTML characters (such as quotes and ampersands).

The full Java regex syntax is described in the following document:

[java.util.regex: Class Pattern](#)

The following table provides sample strings for use with the regex syntax in search criteria.

Search String Entered	Result
a	Adds wildcards before and after your search string (for example, <code>*a*</code>), enabling the search to return results that contain the letter "a".
^a	Adds a wildcard after your search string (for example, <code>a*</code>), enabling the search to return results that begin with the letter "a".
a\$	Adds a wildcard before your search string (for example, <code>*a</code>), enabling the search to return results that end with the character "a".
a*	Searches explicitly for strings containing a character followed by an asterisk (*) for example, "a*".
?	Use a question mark (?) with a character and an asterisk (*) to return zero (0) or more occurrences of a character. For example <code>?a*</code> returns zero or more occurrences of the character "a".

- **newValue** — Specifies the replacement text.
- **ignoreCase** — Ignores case when set to true, but becomes case-sensitive when set to false. The default value is false.

Example XML File for Searching for and Replacing Text Strings

The following is a partial example of an XML file for searching for and replacing a text string.

```
<?xml version="1.0" encoding="utf-8"?>
<actions>
<action command="textReplace" oldValue="boots" newValue="HoleyShoes"
ignoreCase="true"/>
</actions>
```

Searching for and Replacing Multiple Catalog Text Strings

Use the following procedure to search for and replace multiple catalog text strings all at the same time.

1. Create the XML file for searching for and replacing multiple text strings.
See [About Searching for and Replacing Multiple Catalog Text Strings](#).
2. In Catalog Manager, open the catalog in offline mode.
3. From the **Tools** menu, select **XML Search and Replace**.
4. In the **Import from File** field, enter the path or click **Browse** to specify the XML file that you created in Step 1.
5. To make the search case-sensitive, select the **Case Sensitive** box.
6. Click **OK**.

Creating Reports to Display Catalog Data Using Catalog Manager

You can create reports to display catalog data for all catalog object types. You can either display the report on the screen or save it to a file.

When you create a report, a blank or empty field is exported as a tab character. If you create a report with the default of a tab as the field separator, then two tab characters in the report file indicate a blank field.

1. In Catalog Manager, open the catalog. To create a report that shows the SQL statement that is sent to the Oracle BI Server for the object, open the catalog in online mode.
2. Select the top folder for the catalog.
3. From the **Tools** menu, select **Create Report**.
4. Select the catalog object type for which you want to create a report.
5. To eliminate any rows that are the same from the report, select the **Distinct** box.
6. Specify the columns to be displayed in the report in the Columns in Report list. Use the left and right-arrow buttons (< and >) to move the columns between the Available Columns list and the Columns in Report list, and the plus and minus buttons (+ and -) to set the order in which columns are displayed in the report.
7. Click **OK**.
8. Repeat Steps 4 through 7 until the report contains the appropriate columns.
9. To save the report to a file, in the **Save report to** field, specify the path name of the file. Click the **Browse** button to display the Save As dialog for selecting the path name (if the file does not exist, then it is created).
10. Select **Excel Format** to specify to create a file with a .tab extension that can be imported into Microsoft Excel.

For details on supported Excel versions as part of Microsoft Office, see the system requirements and certification documentation [System Requirements and Certification](#).

11. Click **OK**.

Sample Uses for Reports

Running reports can not only help you maintain data within the system, but also can help identify issues before they become problematic.

You can generate reports for various purposes, as described in the following examples:

- To see which dashboards are using an analysis, you can run a Dashboard report including analyses, and search that report for the analysis
- To find analyses that are affected by a changed column in a repository table, you can run an Analysis report that includes all columns and formulas, and then search the report for the items that must then be replaced in Catalog Manager.
- You can create a report that displays all the dashboard prompts and related fields (such as column, formula, and subject area) within the dashboards. You can also create a report of analyses and extract the filters that are used within those analyses. The following is an example of extracting filters in which the formula is derived using a saved filter that is prompted:

Example: "Markets"."Region" [Filter, prompted]

- You can create a report that displays the ACLs for objects. By reviewing the ACLs in the report, you can verify that access to objects is granted to the proper roles with the proper permissions, such as Read/Write. The following line shows an example of ACLs in the report:

"^biconsumer=RX:steve=F", where the caret (^) indicates an application role and "nothing" indicates a user.

Archiving and Unarchiving Using Catalog Manager

Catalog Manager provides the ability to archive and unarchive either an individual catalog folder or an entire catalog.

See the following list for important information on this functionality:

- When you archive an individual catalog folder, all objects in the folder and the folder's subfolders are saved in single compressed file. Properties and attributes of objects are included in the archive file.
- When you unarchive an individual catalog folder, the archive file is uncompressed and all objects in the folder and the folder's subfolders are then stored in the current offline catalog. Existing folders that have the same names as folders being unarchived are overwritten without warning.
- Archiving and unarchiving an entire catalog using Catalog Manager or the Catalog page in Presentation Services is not recommended. Instead, use either the 7-Zip utility or .tar.gz files on UNIX systems for archiving and unarchiving an entire catalog.
- When you unarchive with the Preserve ACL option, application roles and users that are part of ACL entries in the archive must exist on the receiving system. Without these application roles and users, the respective ACL entries aren't preserved on the receiving system.

- Do not archive an entire catalog by starting at the root level (\). Always specify specific folder names when archiving.
- Do not use the following utilities when archiving and unarchiving an entire catalog:
 - WinZip — The WinZip utility does not always handle extended UNIX file permissions properly. Using the WinZip utility to move catalogs in heterogeneous environments might lead to corrupted catalog files.
 - FTP — Some third-party FTP programs rename internal files, which might lead to corrupted catalog files. Before moving a catalog using an FTP program, use the 7-Zip utility to compress the catalog into one file.

You can use the Catalog page in Presentation Services to archive and unarchive objects. For information, see Archiving Objects in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

Archiving a Folder Using Catalog Manager

Use the following procedure to archive a catalog folder.

1. In Catalog Manager, open the catalog in offline mode.
2. Highlight the catalog folder and from the **File** menu, select **Archive**.
3. In the **Archive File Path** field, specify the path name of the file in which to archive the folder. Click **Browse** to display a dialog for selecting the path name.
4. To archive the folder:

- Timestamps that are assigned to the objects and folders that you are archiving, then select the **Keep file time stamps** option.

If you do not select this option, then the archiving process does not include timestamp information and the **Old** option in the Paste Overwrite area of the Preferences dialog is ignored. Upon unarchiving, the system applies a timestamp that indicates the time at which the object or folder is unarchived.

See [Advanced Options for Pasting Objects](#).

- Permissions that are assigned to each object or folder, then select the **Keep permissions** option.

If you do not select this option, then the archiving process does not include any permissions and the options in the Paste ACL area of the Preferences dialog are ignored. Upon unarchiving, the system assigns the parent folder's permissions to all of the objects and folders.

5. Click **OK**.

Unarchiving a Folder Using Catalog Manager

You can use Catalog Manager to remove a folder from archive.

Unarchiving is similar to pasting and therefore requires that you understand the issues that relate to permissions and ACLS as described in [Advanced Options for Pasting Objects](#). Use the following procedure to unarchive a catalog folder.

1. In Catalog Manager, open the catalog in offline mode.
2. To unarchive a catalog folder, navigate to the location where you want to unarchive the folder.

3. From the **File** menu, select **Unarchive**.
4. In the **Archive File Path** field, specify the path name of the catalog folder to unarchive. Click **Browse** to display a dialog for selecting the path name.
5. Click **OK**.

Part VI

Advanced Configuration Settings

Part V describes configuration settings that are required for deploying a system.

This part describes advanced configuration settings that are not required but are optional and advanced settings for fine-tuning a deployment.

This part includes the following chapters:

- [Configuring and Managing Analyses and Dashboards](#)
- [Configuring and Managing Agents](#)
- [Configuring Advanced Options for Mapping and Spatial Information](#)
- [Configuring Resource Availability and URL Generation](#)

18

Configuring and Managing Analyses and Dashboards

This chapter describes how to configure and manage analyses and dashboards and the objects that they contain, such as views, in Oracle Business Intelligence. For information about how content designers work with analyses and dashboards, see *User's Guide for Oracle Business Intelligence Enterprise Edition*.

End users with appropriate privileges can modify personal and shared dashboards, including the addition of pages and content. End users cannot create analyses and dashboards.

This chapter includes the following sections:

- [Managing Dashboards](#)
- [Performing General Configuration Tasks for Analyses](#)
- [Configuring for Displaying and Processing Data in Views](#)
- [Configuring for Prompts](#)
- [Manually Changing Presentation Settings](#)
- [Blocking Analyses in Answers](#)
- [Specifying View Defaults for Analyses and Dashboards](#)
- [Configuring for Write Back in Analyses and Dashboards](#)
- [Customizing the Oracle BI Web User Interface](#)
- [Embedding External Content in Dashboards](#)

Managing Dashboards

Before you create shared dashboards, ensure that you have planned the Oracle BI 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 Oracle BI Presentation Catalog structure and security framework, are provided in Controlling Access to Saved Customization Options in Dashboards in *Security Guide for Oracle Business Intelligence Enterprise Edition*.
- Information about shared folder structures in the Oracle BI Presentation Catalog is provided in [Configuring and Managing the Oracle BI Presentation Catalog](#).

- Information about permissions is provided in *Security Guide for Oracle Business Intelligence Enterprise Edition*.
- Details for enabling users to act for others, which allows them to access the other users' dashboards, is provided in Enabling Users to Act for Others in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

Performing 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:

- [Increasing Heap Size to Assist in Exports to Excel](#)
- [Manually Configuring for Export](#)
- [Supporting Nested Folders, Navigation, and Drill-Down](#)

Increasing 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*. While users can export directly to an Excel format, they might notice greater performance when exporting large numbers of rows if they export first to CSV, then import that file into Excel.

If a user exports a large data set 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.

1. Open the `objh.properties` file for editing. You can find the file at:

```
ORACLE_HOME/bi/modules/oracle.bi.cam.objh/env/  
objh.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 [Using the JavaHost Service for Oracle BI Presentation Services](#).

5. Save and close the file.
6. Restart Oracle Business Intelligence.

Manually Configuring 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)
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>
  </Download>
</ServerInstance>
```

```
</Export>

</Download>
</ServerInstance>
```

5. Save your changes and close the file.
6. Restart Oracle Business Intelligence.

The following table describes the elements for manually configuring for export.

Element	Description	Default Value
DataValue	<p>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.</p> <p>Valid values are:</p> <ul style="list-style-type: none"> • UseRawValue • UseFormattedValue <p>The export type is:</p> <pre>xsi:type="excel"</pre>	UseRawValue
RepeatRows	<p>Specifies whether cells that span rows and cells that span columns are to be repeated when exporting tables and pivot tables to Excel.</p> <p>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.</p> <p>If set to false, then the behavior is the same as that defined by the Value Suppression option in the Analysis editor:</p> <ul style="list-style-type: none"> • 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. <p>For more information on the Value Suppression option, see <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i></p> <p>The export type is:</p> <pre>xsi:type="excel"</pre>	false
Delimiter	<p>Specifies the column delimiter character for the CSV Format option, for example, a semicolon (;), when exporting raw data from results and views.</p> <p>The export type is:</p> <pre>xsi:type="formattedText"</pre>	","

Element	Description	Default Value
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: <code>xsi:type="pdf"</code>	false
Orientation	Specifies the orientation (either Portrait or Landscape) when exporting to PDF and to Powerpoint. The export type for PDF exports is: <code>xsi:type="pdf"</code> The export type for Powerpoint exports is: <code>xsi:type="ppt"</code>	Landscape (for PDF exports) Portrait (for Powerpoint exports)
QuoteTxtTab	Adds quotes for the CSV Format option. When set to false, no quotes are added. The export type is: <code>xsi:type="formattedText"</code>	true

Supporting Nested Folders, Navigation, and Drill-Down

The administrator can set up subject areas in ways that assist content designers who work with analyses.

Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition 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 Oracle BI 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.

Configuring for Displaying and Processing Data in Views

You can configure various options that change the display and processing of data in views.

See also [Using Fusion Middleware Control to Set Configuration Options for Data in Tables and Pivot Tables](#) and [Using Fusion Middleware Control to Set the Maximum Number of Rows Processed to Render a Table View](#).

This section contains the following topics:

- [Manually Configuring for Data in Views](#)
- [Manually Configuring for Graphs and Gauges](#)
- [Manually Changing Alternating Bar Color](#)
- [Manually Configuring for Interactions in Views](#)

Manually Configuring for Data in Views

You can configure various options that change the processing and display of data in views.

See the following sections:

- [Manually Configuring Cube Settings for Pivot Tables and Graphs](#)
- [Manually Configuring Settings for Data in Views](#)
- [Manually Configuring Settings for Fetching Data for Table Views, Pivot Table Views, and Trellis Views](#)

Manually Configuring 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 element:
 - `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 do not count. The default is 40000.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<ServerInstance>
  <Views>
    <Cube>
      <CubeMaxRecords>40000</CubeMaxRecords>
    </Cube>
  </Views>
</ServerInstance>
```

 **Note:**

Both CubeMaxRecords and ResultRowLimit limit the number of rows returned. The limit is determined by the setting with the larger value (see [Using 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 Business Intelligence.

Manually Configuring 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>
      <MaxVisibleRows>500</MaxVisibleRows>
      <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>
      <MaxVisibleRows>500</MaxVisibleRows>
      <MaxVisibleSections>25</MaxVisibleSections>
      <DefaultRowsDisplayed>30</DefaultRowsDisplayed>
      <DefaultRowsDisplayedInDelivery>250</DefaultRowsDisplayedInDelivery>
      <DefaultRowsDisplayedInDownload>65000</DefaultRowsDisplayedInDownload>
    </Pivot>
    <Trellis>
      <Simple>
```



```

        <MaxCells>1000</MaxCells>
        <MaxVisibleSections>10</MaxVisibleSections>
        <MaxVisiblePages>1000</MaxVisiblePages>
        <MaxVisibleRows>100</MaxVisibleRows>
        <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>
        <MaxVisibleRows>250</MaxVisibleRows>
        <MaxVisibleColumns>150</MaxVisibleColumns>
        <MaxPagesToRollOutInDelivery>1000</MaxPagesToRollOutInDelivery>
        <DefaultRowsDisplayed>25</DefaultRowsDisplayed>
        <DefaultRowsDisplayedInDelivery>250</DefaultRowsDisplayedInDelivery>
        <DefaultRowsDisplayedInDownload>10000</
DefaultRowsDisplayedInDownload>
    </Advanced>
</Trellis>
<Charts>
    <MaxVisibleColumns>2000</MaxVisibleColumns>
    <MaxVisiblePages>1000</MaxVisiblePages>
    <MaxVisibleRows>2000</MaxVisibleRows>
    <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>
    <MaxVisibleRows>10000</MaxVisibleRows>
    <MaxVisibleSections>50</MaxVisibleSections>
</Treemap>
</Views>
</ServerInstance>

```

Note that this example does not include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and cannot be changed manually.

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

The table below describes the common elements that affect the display of data in views. If the 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 should not exceed the number that is specified for the MaxVisibleRows element.	25 (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 is 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 is 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 is downloaded to a CSV file.	65000	Table
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 should not 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 is displayed on a dashboard.	1000	Pivot Table, Table, Trellis

Element	Description	Default Value	Applicable Views
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)	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
MaxVisibleRows	Specifies the maximum number of rows to be displayed in a view. The value of DefaultRowsDisplayed should not exceed this value. For tables and pivot tables, specifies the number of rows that is displayed on the tooltip for the Display Maximum Rows per Page paging control button.	2000 (100 for Simple Trellis; 250 for Advanced Trellis; 10000 for Treemap)	Graph, Pivot Table, Table, Trellis, Treemap
MaxVisibleSections	Specifies the maximum number of sections to be displayed in a view. This element does not 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 Configuring for 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 is sent to the browser for a single graph.	4096	Graph

Manually Configuring Settings for Fetching 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 be used 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*.

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 Business Intelligence.

The table below 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 is 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
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

Element	Description	Default Value
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: <ul style="list-style-type: none"> 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 Configuring for 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 Configuring 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 Business Intelligence.

Element	Description	Default Value
EmbedFonts	See Configuring Fonts for Graphs for details.	false
DefaultWebImageType	<p>Specifies the default type for rendering an image when a format has not been specified in the URL or in the XML file for the view. Valid values are:</p> <ul style="list-style-type: none"> • flash In a browser that does not support the flash format, the image does not render. You should use the html5 value instead. • png (W3C Portable Network Graphics) • svg (W3C Scalable Vector Graphics) The svg value is not supported in this release, so flash is used if svg is specified. • html5 In a browser that does not support the html5 format, the image renders in the flash format instead. <p>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.</p>	flash
SectionSliderDefault	<p>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.</p> <p>For more information about defining section sliders in graphs, gauges, and funnels, see <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i>.</p>	5
SectionSliderLimit	Specifies the maximum number of values that can be displayed on a section slider bar.	10
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

Element	Description	Default Value
FlashCLSID	<p>Specifies a custom global identifier (clsid) property for downloading Flash.</p> <p>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 Business Intelligence 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.</p> <p>Test flash graphs independent of Oracle Business Intelligence to ensure that they function with the custom global identifier property.</p>	No default value

Configuring Fonts for Graphs

There are two ways you can configure fonts for graphs.

- Set the embed fonts element
- Deliver font files for printing

Setting 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.

Delivering 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.

Configuring Graph and Gauge Rendering

You can use the `DefaultWebImageType` element to specify the default type for rendering an image when a format has not been specified in the URL or in the XML file for the view.

Valid values are:

- flash

In browsers that do not support flash format, the graph or gauge will not render. You should use the `html5` value instead.

- png (W3C Portable Network Graphics)
- svg (W3C Scalable Vector Graphics)

The `svg` value is not supported in this release, so `flash` is used if `svg` is specified.

- html5

In browsers that support only the flash format, the graph or gauge will render 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 Changing 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 Configuring 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.

The elements in the `instanceconfig.xml` file specify the default settings for a new or upgraded 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 Business Intelligence.

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

Configuring for Prompts

You can configure settings that affect the way that users work with prompts.

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>
    </AutoCompletePromptDropDowns>
  </Prompts>
</ServerInstance>
```

```

        <ResultsLimit>50</ResultsLimit>
    </AutoCompletePromptDropDowns>
    <ShowNullValueWhenColumnIsNullable>never</ShowNullValueWhenColumnIsNullable>
</Prompts>
</ServerInstance>

```

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Element	Description	Default Value
AutoApplyDashboardPromptValues	<p>Specifies whether to display various fields, as described in the following list:</p> <p>If true, then</p> <ul style="list-style-type: none"> • 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. <p>If false, then</p> <ul style="list-style-type: none"> • The Show Apply button and Show Reset button fields are not displayed on the Edit Page Settings dialog. • The Prompts Apply Buttons and Prompts Reset Buttons fields are not displayed in the Dashboard Properties dialog. • The Prompt Buttons on Current Page option is not 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
CaseInsensitive	Specifies whether the auto-complete functionality is case-insensitive. If set to true, case is not 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

Element	Description	Default Value
Matching Level	<p>Specifies whether the auto-complete functionality uses matching to find the prompt value that the user enters into the prompt field. These settings do not apply when the user accesses the Search dialog to locate and specify a prompt value.</p> <p>Use the following settings:</p> <ul style="list-style-type: none"> • 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	<p>Specifies the maximum number of choices to display in the following locations:</p> <ul style="list-style-type: none"> • 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 is 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 is 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 is 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
ShowNullValueInPromptsWhenDatabaseColumnIs Nullable	<p>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.</p> <p>Use the following settings:</p> <ul style="list-style-type: none"> • 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	<p>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.</p> <p>A setting of false turns auto-complete off, which means that the auto-complete fields in the My Account and Dashboard Properties dialogs are not available.</p>	false, unless you are running Oracle BI EE on the Oracle Exalytics In-Memory Machine

Manually Changing Presentation Settings

You can configure settings that change the display of dashboards and presentation settings.

See the following sections:

- [Manually Changing Presentation Setting Defaults](#)
- [Providing Custom Links in Presentation Services](#)
- [Enabling the Ability to Create Links to Dashboard Pages](#)
- [Configuring an Alternate Toolbar for Oracle BI Publisher](#)
- [Enabling the Ability to Export Dashboard Pages to Oracle BI Publisher](#)
- [Modifying the Table of Contents for PDF Versions of Briefing Books](#)
- [Configuring a Custom Download Link for the Smart View Installer](#)

Manually Changing 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>
  <AnalysisEditorStartTab>answerResults</AnalysisEditorStartTab>
  <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 does not include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and cannot be changed manually.

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Element	Description	Default Value
AnalysisEditorStartTab	Specifies whether the Analysis editor is to open by default to the Criteria tab or the Results tab. This setting is applied when users click an analysis' Edit link from a dashboard, the Home Page, or the Catalog page. Valid values are: <ul style="list-style-type: none"> • answerResults • answerCriteria Users can override this default setting by setting the Full Editor option in the My Account dialog.	answerResults
Enable508	Specifies whether content for Oracle BI EE is rendered in a browser in a way that facilitates the use of a screen reader. If you set this to true, then the BI Composer wizard in accessibility mode is used as the analysis editor, regardless of the setting of the Analysis Editor component. If you set this to false, and the setting of the Analysis Editor component is Wizard (limited functionality) , then the BI Composer wizard in regular mode is used as the analysis editor. Users can override this default setting by setting the Accessibility Mode option in the Sign In page or the My Accounts dialog.	false

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 BI EE searches for dashboard template pages. By default, Oracle BI EE 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. See <i>Delaying the Execution of Dashboard Pages in User's Guide for Oracle Business Intelligence Enterprise Edition</i> .	true
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 do not 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 Modifying the Table of Contents for PDF Versions of Briefing Books . For information about working with briefing books, see <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i> .	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 is not displayed appropriately. Use this setting to set the maximum number of significant digits, such as 16 on Linux platforms.	No default value



Note:

See [Enabling the Ability to Create Links to Dashboard Pages](#) for information about the defaults for the Bookmarks, MaxAgeMinutes, EnableBookmarkURL, and EnablePromptedURL elements. See [Configuring an Alternate Toolbar for Oracle BI Publisher](#) for information about the defaults for the ReportingToolbarMode element.

Providing Custom Links in Presentation Services

By default, the global header in Oracle BI EE contains menus and options that enable you to navigate easily among features. You might like to customize the global header and the Get Started section of the Home page to better meet the needs of users by disabling certain links or including your own links.

Changes that you make to the Get Started section do not affect the Help menu in the global header. For 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.

To customize the global header, you perform the tasks that are described in the following sections:

- [Creating the customlinks.xml File](#)
- [Adding the CustomLinks Element](#)
- [Setting the Custom Links Privilege](#)

Creating 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 [Adding 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.

Table 18-1 Elements in `customlinks.xml`

Element or Attribute	Optional?	Data Type	Description
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.
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. getstarted: Specifies to include the link in the Get Started section of the Home page.
location: insertBefore	Optional	String	Specifies the ID of an existing link before you which you want to insert this link. If the specified ID is invalid, then the link is inserted in the default locations. See Specifying the insertBefore Attribute for information on valid values.
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: name	Required	String	Specifies the name of the link that is not translated.
link: localizedName	Optional	String	Specifies the message ID of the link that is translated, which takes precedence over the non-translated name.
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. See <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i> for information on the fmap syntax.
link: iconLarge	Optional	String	Specifies the file name of an icon to display with the link in the Get Started section. The display of icons is controlled by the fmap syntax.
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']&&& privileges.Access['Global Delivers']
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 releases, the vpat attribute served the same purpose as the accessibility attribute. The vpat attribute has been deprecated.
link: src	Required	String	Specifies the URL for the link.

Table 18-1 (Cont.) Elements in `customlinks.xml`

Element or Attribute	Optional?	Data Type	Description
link: target	Optional	String	Specifies the browser window in which to open the link. The values are: self : Opens in same window in which Presentation Services is running. blank : Opens in a new window. <i>any-name</i> : Opens in a window with the specified name.
link: description	Optional	String	Specifies the description of the link that is not translated.
link: localizedDesc	Optional	String	Specifies the message ID of the link that is translated, which takes precedence over the non-translated description.

 **Note:**

You may want to place icons in a common directory structure exposed by the application server or HTTP Server or place them in the catalog file system and expose them via fmap. See *Storing Custom Files Locally and Using the fmap Function to Reference Them* in *User's Guide for Oracle Business Intelligence Enterprise Edition* for information on the FMAP syntax.

After you update the `customlinks.xml` file, the file is reloaded when you next restart Presentation Services, as described in [Managing Oracle Business Intelligence Processes](#).

Specifying the `insertBefore` Attribute

You can include the `insertBefore` attribute that is described in [Table 18-1](#) to specify the ID of an existing link before which you want to insert another link. The following list provides the valid IDs for the global header:

- Navigation Bar
- catalog
- dashboard
- favorites
- home
- new
- open
- user
- Search Bar
- admin
- advanced
- help
- logout

The Get Started section includes no fixed IDs because you can customize its contents.

Sample File and Output

The following code sample shows a `customlinks.xml` file that was edited to include links in the global header and the Get Started section.

```
<?xml version="1.0" encoding="utf-8"?>
<customLinks xmlns="com.siebel.analytics.web/customlinks/v1">
<link id="11" name="OTN" description="OTN open in new window" src="http://
www.oracle.com" target="blank" >
  <locations>
    <location name="header" />
  </locations>
</link>

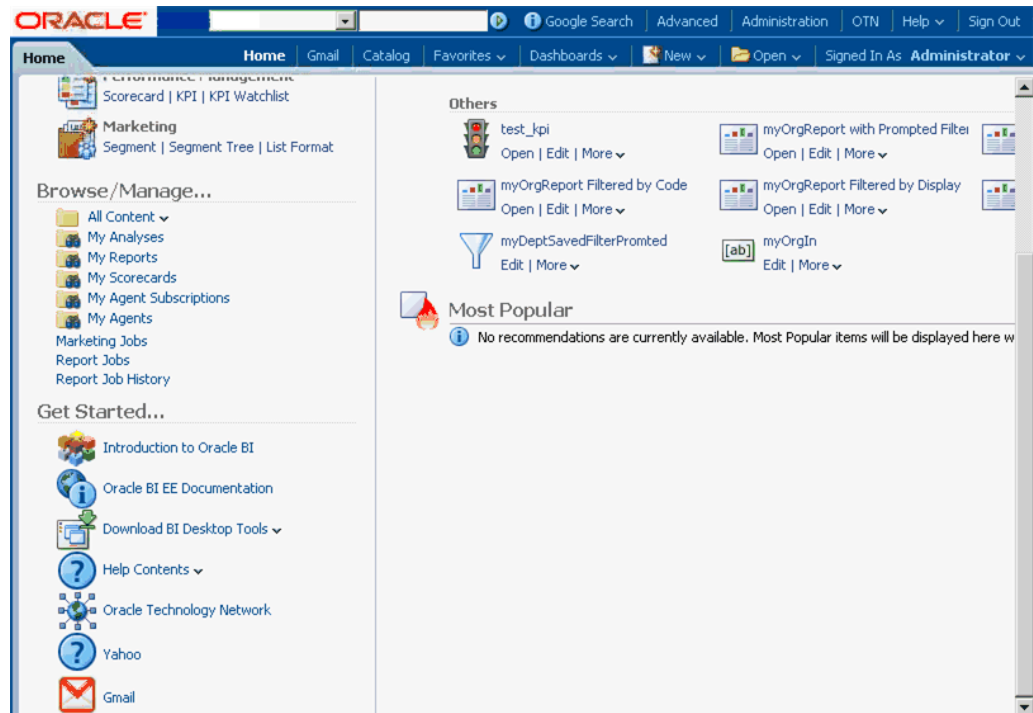
<link id="12" name="Google Search" description="Google open in named window"
src="http://www.google.com/" target="google" iconSmall="common/info_ena.png" >
  <locations>
    <location name="header" insertBefore="advanced" />
  </locations>
</link>

<link id="13" name="Yahoo" description="Yahoo" src="http://www.yahoo.com"
target="yahoo" iconLarge="common/helptopics_lg_qualifier.png">
  <locations>
    <location name="getstarted" />
  </locations>
</link>

<link id="15" name="Gmail" description="gmail" src="http://www.gmail.com"
target="blank" iconLarge="common/gmail.png" >
  <locations>
    <location name="getstarted" />
    <location name="header" insertBefore="catalog" />
  </locations>
</link>
</customLinks>
```

This file modifies the Home page as shown below. Note the following changes to the Home page:

- The global header is modified to include the following:
 - **Google Search** — A link to the Google Home page with a custom icon, which is placed before the **Advanced** link.
 - **OTN** — A link to the Oracle Technology Network page, which is placed before the **Help** link.
 - **Gmail** — A link to the Gmail Home page, which is placed before the **Catalog** link.
- The Get Started section is modified to include links to the Yahoo and Gmail Home pages that use custom icons.



Adding 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.

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 Business Intelligence.

Setting 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 have made, then you must ensure that the Custom Links privilege is assigned to the BI Consumer role, which occurs by default. You cannot 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 Managing Presentation Services Privileges in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

Enabling 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 Oracle BI Presentation Catalog (under the `/system/bookmarks` folder) that captures the state of a dashboard page. It is 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 *User's Guide for Oracle Business Intelligence Enterprise Edition*. For more information on privileges, see Managing Presentation Service Privileges in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

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 (and does not mean that it does not expire). You cannot 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 Business Intelligence.

Configuring an Alternate Toolbar for Oracle BI Publisher

When you include a BI 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 do not require passing of context to or from the BI Publisher report to the larger dashboard-based analytic application, you can display a variant of the default BI Publisher toolbar, which exposes the underlying parameter prompts of that BI Publisher report. Within that frame, a user can then pass in parameters to a single BI Publisher report.

This approach can be confusing to the user as any other dashboard prompts on the page do not contribute to the BI Publisher report, which also does not participate in passing context back to the rest of the application. Changes to the BI Publisher toolbar are also applied globally for all BI Publisher reports that are embedded in dashboards across the entire BI Publisher instance.

Use the `ReportingToolbarMode` element to affect how BI Publisher reports are embedded in Oracle BI EE. You configure the alternate BI 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.

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 BI Publisher toolbar to display the parameter prompts of the BI Publisher report
4. Save your changes and close the file.
 5. Restart Oracle Business Intelligence.

Enabling the Ability to Export Dashboard Pages to Oracle BI 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 BI Publisher. You can enable or disable the ability to export dashboard pages to Oracle BI Publisher by setting the `EnableDashPageExport` element.

See *About Creating Custom Layouts for Printing and Exporting Dashboard Pages in User's Guide for Oracle Business Intelligence Enterprise Edition*.

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 BI 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 BI 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 Business Intelligence.

Modifying 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.

See Working with Briefing Books in *User's Guide for Oracle Business Intelligence Enterprise Edition* for information about the table of contents.

The default template for the table of contents, `toc-template.rtf`, is located in the `ORACLE_INSTANCE\config\OracleBIPresentationServicesComponent\coreapplication_obispn` directory. You can modify the `toc-template.rtf` file to accommodate the needs of your organization.

Configuring a Custom Download Link for the Smart View Installer

The Smart View version packaged with Oracle Business Intelligence may be out of synchronization with the Oracle Business Intelligence 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 BI Desktop Tools list on the Oracle Business Intelligence 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:

11g 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>
```

11g 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>
```

11g 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 Business Intelligence.

Blocking Analyses in Answers

You might want to block specific analyses, such as requiring content designers to include certain columns with others, or requiring filters when certain columns are requested.

Answers includes an API that you can use to block queries based on the criteria specified in the analysis or based on formulas in the analysis. You can access the API using JavaScript to check conditions and validate analyses.

This section contains the following topics:

- [Storing JavaScript Files](#)
- [Blocking Analyses Based on Criteria](#)
- [Blocking Analyses Based on Formula](#)
- [Validation Helper Functions](#)

Storing JavaScript Files

This section explains how to use JavaScript to check conditions and validate analyses.

You write your own JavaScript programs for performing these tasks and other similar ones. Oracle BI EE does not install any JavaScript programs. As you write JavaScript programs, you can store them in the singleton data directory (SDD), which you first deploy as a virtual directory:

```
SDD/components/OBIPS/analyticsRes
```

Where SDD is the Singleton Data Directory for example, *DOMAIN_HOME*/bidata.

See [Setting Up Shared Files and Directories](#).

To place JavaScript programs in a directory other than this one, then you can do so, if you specify the full path name in the code that calls the program. For example, you can use code such as the following:

```
<script type="text/javascript" src="http://example/mydir/myblocking.js" />
```

Blocking Analyses Based on Criteria

When a user attempts to execute an analysis that your code blocks, you can display an error message, and the analysis is not executed.

The answerstemplates.xml file includes a message named `kuiCriteriaBlockingScript` that can be overridden to either define or include JavaScript that defines a `validateAnalysisCriteria` function. By default, this message contains a function that always returns true.

Answers calls your `validateAnalysisCriteria` function when the user tries to execute the analysis. The function can return `true` if the analysis is not blocked, or `false`, or a message if the analysis is blocked. If a message or a value other than `false` is returned, then the message is displayed in a popup window. In either case, the query is blocked.

The following code example shows the blocking of a query. First, place the following XML code in the `answerstemplates.xml` file.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
  <WebMessageTable system="QueryBlocking" table="Messages">
    <WebMessage name="kuiCriteriaBlockingScript" translate="no">
      <HTML>
        <script type="text/javascript" src="fmap:myblocking.js" />
      </HTML>
    </WebMessage>
  </WebMessageTable>
</WebMessageTables>
```

This XML code calls a JavaScript program called `myblocking.js`. Ensure that you place this file in the following directory:

`SDD/components/OBIPS/analyticsRes`

SDD is the Singleton Data Directory.

The following is sample code for the `myblocking.js` program.

```
// 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 you do not override the function using the template as described previously, or if the function returns anything other than `false`, then the criteria is considered to be valid and the analysis is issued. The criteria is validated using this same mechanism for preview and save operations as well.

After making this change, either stop and restart the server for Oracle BI Presentation Services, or click the **Reload Files and Metadata** link on the Administration page.

Blocking Analyses Based on Formula

Answers provides a hook that lets you incorporate a JavaScript validation function that is called from Answers when a content designer enters or modifies a column formula. If the call fails and returns a message, then Answers displays the message and cancels the operation.

Additionally, helper functions are available so the query blocking function can check for filters, columns, and so on, rather than traversing the Document Object Model (DOM) manually. The DOM is a way of describing the internal browser representation of the HTML UI page that is currently being displayed in Answers. See [Validation Helper Functions](#).

The `crieratemplates.xml` file includes a message named `kuiFormulaBlockingScript` that can be overridden to include JavaScript that defines a `validateAnalysisFormula` function. By default, this message contains a function that always returns true.

Answers calls `validateAnalysisFormula` before applying changes made by the content designer. If the function returns true, then the formula is accepted. If the function returns false, then the formula is rejected. Otherwise, the return value from the function is displayed in the message area beneath the formula, as it does currently when an invalid formula is entered.

The content designer has the option to click **OK** to ignore the error. To display your own alert and allow the content designer to continue, your function should return true. To block the query, return false or a message. Your function should investigate the formula passed to it using a JavaScript string and regular expression techniques for validation.

The following code example shows a sample custom message.

```
<?xml version="1.0" encoding="utf-8"?>
<WebMessageTables xmlns:sawm="com.siebel.analytics.web.messageSystem">
  <WebMessageTable system="QueryBlocking" table="Messages">
    <WebMessage name="kuiFormulaBlockingScript" translate="no">
      <HTML>
        <script type="text/javascript" src="fmap:myblocking.js" />
      </HTML>
    </WebMessage>
  </WebMessageTable>
</WebMessageTables>
```

The following code example shows blocking based on the formula entered.

```
// This is a formula blocking function. It makes sure the user does not enter an
// unacceptable formula.
function validateAnalysisFormula(sFormula, sAggRule)
{
  // do not 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 is
    not allowed.";
  // no case statements
  var caseRe = /CASE.+END/gi;
```

```

    if (sFormula.search(caseRe) >= 0)
        return "Do not use a case statement.";
    // Check for a function syntax: aggrule(formula) aggrule should not contain a '.'
    var castRe = /^\[s*\w+\s*\(.+\)\s*\$/gi;
    if (sFormula.search(castRe) >= 0)
        return "Do not use a function syntax such as RANK() or SUM().";
    return true;
}

```

After making this change, either stop and restart the server for Oracle BI Presentation Services, or click the **Reload Files and Metadata** link on the Administration page.

Validation Helper Functions

Validation helper functions are defined in a JavaScript file.

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
<code>CriteriaValidator.getSubjectArea()</code>	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.
<code>CriteriaValidator.tableExists(sTable)</code>	Returns true if the specified folder (table) has been added to the analysis by the content designer, and false if the folder was not added.
<code>CriteriaValidator.columnExists(sTable, sColumn)</code>	Returns true if the specified column has been added to the analysis by the content designer, and false if the column was not added.
<code>CriteriaValidator.dependentColumnExists(sCheckTable, sCheckColumn, sDependentTable, sDependentColumn)</code>	Checks to ensure that the dependentColumn exists if the checkColumn is present. It returns true if either the checkColumn is not 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.
<code>CriteriaValidator.filterExists(sFilterTable, sFilterColumn)</code>	Returns true if a filter exists on the specified column, and false if no filter is present.
<code>CriteriaValidator.dependentFilterExists(sCheckTable, sCheckColumn, sFilterTable, sFilterColumn)</code>	Checks to ensure that the dependentFilter exists if the checkColumn is present in the projection list. It returns true if either the checkColumn is not present, or the checkColumn and the dependent filter are present.
<code>CriteriaValidator.filterCount(sFilterTable, sFilterColumn)</code>	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 is not 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 cannot be determined.

Specifying 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 cannot modify its contents.

The wrapper message adds the combined XML into a JavaScript variable, `kuiDefaultViewElementsXML`, that is 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.

- [Adding a Default Header or Footer to New Analyses](#)
- [Preventing Auto-Previewing of Results](#)
- [Setting Defaults for Analyses in the Compound Layout](#)
- [Changing Dashboards Section Defaults](#)
- [Specifying Dashboard Page Defaults Including 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.

Adding 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 type="top"><caption>[b]Acme Confidential[/b]</caption>
          <displayFormat fontColor="#FF0000"/></zone>
        </pageFooter>

      </pageProps>
      <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>
```

Preventing Auto-Previewing 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>

```

Setting 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>

```

Changing 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>

```

Specifying Dashboard Page Defaults Including 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>

```


Configuring 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:

- [How Write Back Works](#)
- [Process for Configuring Write Back](#)
- [Example: Process for Configuring Write Back](#)
- [Write-Back Limitations](#)
- [Creating Write-Back Template Files](#)
- [Setting the LightWriteback Element](#)

How Write Back Works

If a user has the Write Back to Database privilege, then the write-back fields in analyses can display as editable fields if properly configured.

If the user does not have this privilege, then the write-back fields display as normal fields. If the user types a value in an editable field and clicks the appropriate write-back button, then the application reads the write-back template to get the appropriate insert or update SQL command. It then issues the insert or update command. If the command succeeds, then it reads the record and updates the analysis. If there is an error in either reading the template or in executing the SQL command, then an error message is displayed.

The insert command runs when a record does not yet exist and the user enters new data into the table. In this case, a user has typed in a table record whose value was originally null.

The update command runs when a user modifies existing data. To display a record that does not yet exist in the physical table to which a user is writing back, you can create another similar table. Use this similar table to display placeholder records that a user can modify in dashboards.



Note:

If a logged-on user is already viewing a dashboard that contains an analysis where data has been modified using write back, the data is not automatically refreshed in the dashboard. To see the updated data, the user must manually refresh the dashboard.

Process for Configuring Write Back

Follow the steps below to configure write back.

1. (for the content designer) Work with the Oracle BI Administrator of the repository to assess the reporting needs in the organization and make a list of write-back columns that are needed and the analyses in which they are displayed.

Hierarchical columns do not support the write-back capability but attribute columns, measure columns, and double columns do support the write-back capability. For double columns, you can write back to the display column. No automatic translation of the code column is provided.

2. (for the Oracle BI Administrator of the repository) Create a physical table in the database that has a column for each write-back column needed and select the **Allow direct database requests by default** option in the General tab of the Database dialog.



Note:

For optimum security, store write-back database tables in a unique database instance.

See *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*

3. (for the Oracle BI Administrator of the repository) Enable write back on the columns using the Oracle BI Administration Tool. This includes:
 - Disabling caching on the physical table
 - Making the logical columns writeable
 - Enabling **Read/Write** permissions for the presentation columns

See *Enabling Write Back On Columns in Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*

4. (for the administrator) Create write-back template files that specify the SQL commands that are necessary to both insert and update values into table views for which you want to enable write back.

For how, see [Creating Write-Back Template Files](#).

5. (for the administrator) Add the **LightWriteback** element in the instanceconfig.xml file.

See [Setting the LightWriteback Element](#).

6. (for the administrator) In Oracle BI Presentation Services, grant the following write-back privileges to the appropriate users: Manage Write Back and Write Back to Database.

For how, see *Managing Presentation Services Privileges in Security Guide for Oracle Business Intelligence Enterprise Edition*.

7. (for the content designer) Add the write-back capability to columns.

See *Adding the Write-Back Capability to a Column in User's Guide for Oracle Business Intelligence Enterprise Edition*.

8. (for the content designer) Add the write-back capability to table views.

See *Adding the Write-Back Capability to a Table View in User's Guide for Oracle Business Intelligence Enterprise Edition*.

Example: Process for Configuring Write Back

Refer to the following example which demonstrates how the Oracle BI Administrator works with the content designer to configure write back processes.

- After working with the Oracle BI Administrator of the repository, the content designer determines that:
 - These write-back columns are needed: YR, Quarter, Region, ItemType, and Dollars
 - The analysis in which these columns are displayed is titled Region Quota
- The Oracle BI Administrator of the repository does the following:
 - Creates a physical table called regiontypequota that includes the YR, Quarter, Region, ItemType, and Dollars columns
 - Selects the **Allow direct database request by default** option in the General tab of the Database dialog
- Using the Oracle BI Administration Tool, the Oracle BI Administrator of the repository:
 - Makes the WriteBack table noncacheable in the Physical layer
 - Makes the Dollars column in the WriteBack table writeable in the Business Model layer
 - Enables Read/Write permission on the Dollars column in the WriteBack table in the Presentation layer to the BI Author and Authenticated User
- The administrator creates a write-back template file that contains the following write-back template, named SetQuotaUseID:

```
<?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(@{c0},{c1},'@{c2}','@{c3}','@{c4}')</insert>
        <update>UPDATE regiontypequota SET Dollars=@{c4} WHERE YR=@{c0} AND
Quarter=@{c1} AND Region='@{c2}' AND ItemType='@{c3}'</update>
      </writeBack>
    </XML>
  </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

And stores it in:

SDD/components/OBIPS/custommessages

Where SDD is the Singleton Data Directory for example, DOMAIN_HOME/bidata.

- The administrator adds the **LightWriteback** element to the instance.config.xml file as follows:

```
<WebConfig>
  <ServerInstance>
    <LightWriteback>true</LightWriteback>
```

```
</ServerInstance>
</WebConfig>
```

6. The administrator grants the following privileges using the Administration: Manage Privileges page in Oracle BI Presentation Services:
 - Manage Write Back to the BI Author
 - Write Back to Database to the Authenticated User
7. The content designer edits the Region Quota analysis and, for each of the following columns, enables the column for write back by completing the Column Properties dialog: Write Back tab:
 - YR
 - Quarter
 - Region
 - ItemType
 - Dollars
8. The content designer edits the table view in the Region Quota analysis and enables it for write back by selecting the **Enable Write Back** box and entering **SetQuotaUseID** in the **Template Name** field in the Table Properties dialog: Write Back tab.

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 the Oracle BI Server. As you configure for write back, keep the following limitations in mind:

- Numeric columns must contain numbers only. They should not contain any data formatting characters such as dollar signs (\$), pound signs or hash signs (#), percent signs (%), and so on.
- Text columns should 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 is not 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 is not 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 can cause problems.
- Any field in an analysis can be flagged as a write-back field, even if it is not derived from the write-back table that you created. However you cannot successfully execute the write-back operation if the table is not 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 does not 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 does not 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 executed, 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 is not allowed."
- The template mechanism is not suitable for entering arbitrary new records. In other words, do not use it as a data input tool.
- When creating a table for write back, ensure that at least one column does not include write-back capability but does include values that are unique for each row and are non-null.
- Write-back analyses do not support drill-down. Because drilling down modifies the table structure, the write-back template does not 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.

Creating 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 have 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.

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 is used for configuring a write-back template is usually not translated, because it is 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:

- You must specify a name for the write-back template using the `name` attribute of 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.

The following example shows the specification of the write-back template that is called "SetQuotaUseID."

```
<WebMessage name="SetQuotaUseID">
```

- 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 can be referenced either by column position (such as `@1`, `@3`) or by column ID (such as `@{c1234abc}`, `@{c687dfg}`). Column positions start numbering with 1. The use of column ID is preferred. Each column ID is alphanumeric, randomly generated, and found in the XML definition of the analysis in the Advanced tab of the Analysis editor.
- You must include both an `<insert>` and an `<update>` element in the template. If you do not 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 cannot read the Write Back Template 'my_template'".

- If a parameter's data type is not an integer or real number, then add single quotation marks around it. If the database does not 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 18-1 Examples: Write-Back Template Files

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(@{c0},{c1},{c2},{c3},{c4})</insert>
        <update>UPDATE regiontypequota SET Dollars=@{c4} WHERE YR=@{c0} AND
Quarter=@{c1} AND Region=@{c2} AND ItemType=@{c3}</update>
      </writeBack>
    </XML>
  </WebMessage>
</WebMessageTable>
</WebMessageTables>
```

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>
```

Setting 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 does not include elements that might exist in the file, but that are centrally managed by Fusion Middleware Control and cannot be changed manually.

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Customizing the Oracle BI Web User Interface

The user interface in Oracle BI EE is generated by using scripts and is therefore highly customizable.

The look-and-feel is controlled by skins and styles. Skins define the user interface chrome (visible graphic features) outside the home and dashboard area.

Oracle BI EE is shipped with several styles, such as Skyros, blafp (browser look and feel), and FusionFX (fusion applications).

The following sections provide information about how to customize the web user interface:

- [What Are Skins and Styles?](#)
- [General Tips for Customizing the Web User Interface](#)
- [About Style Customizations](#)
- [Modifying the User Interface Styles for Presentation Services](#)
- [Customizing Your Style](#)
- [Example of Modifying the Skyros Master Branding Class](#)

What Are Skins and Styles?

Skins and styles change the way an interface looks.

You can control the way that the interface for Oracle BI EE is displayed to users by creating skins and styles. The primary difference between skins and styles is that styles only apply to dashboard content, whereas skins apply to every other part of the user interface. For example, within Oracle BI EE, skins apply to components that are used in analyses and scorecarding.

You specify the default style and skin 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 [Modifying the User Interface Styles for Presentation Services](#) for additional information.

Styles and skins 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](#) for additional information.

General Tips for Customizing the Web User Interface

Some points to keep in mind as you plan to customize the web user interface.

- In Oracle BI Presentation Services, you customize the user interface elements and appearance by modifying skins, and styles, not with JavaScript, and you do not 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. See *Working with HTML Markup in User's Guide for Oracle Business Intelligence Enterprise Edition*.

- You can adapt the Oracle BI EE deployment to a particular language. See [Localizing Oracle Business Intelligence](#).

About Style Customizations

Oracle BI EE ships with various styles, such as blaf (browser look and feel), FusionFX (fusion applications), and Skyros. If you are going to customize the look-and-feel of Oracle BI EE or Oracle BI Publisher, Oracle strongly recommends that you use the custom style provided in the bicustom-template.ear as your starting point.

This custom style is a copy of the Skyros style. See [Modifying the User Interface Styles for Presentation Services](#) for additional information.

Most of the common Skyros styles and image files, including the style sheet (master.css), are contained in a master directory. The topic [Customizing Your Style](#) describes this directory and its structure in detail.

Within the style sheet, each element (or class) that is available for update is documented by comments.

Other style sheets are also contained within the Skyros style and skin folders. You are not likely to need to update these files unless you are creating an advanced custom skin that provides styles for each detail of the user interface.

Modifying the User Interface Styles for Presentation Services

When creating your own styles and skins for Presentation Services, you must create CSS, graph.xml, and image files, and then make them available to Oracle BI EE. 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: Deploying the "bicustom.ear" File for the First Time?](#)
To redeploy the bicustom.ear file, see [Approach 1: Redeploying the "bicustom.ear" File](#).
- **Approach 2:** You use a shared file system. This approach is useful when you want to see your changes in Oracle BI EE immediately or when your customizations are outside the Presentation Services firewall. See [Approach 2: Deploying Using Shared Folders](#).

To view the modifications that you made using Approach 2, see [Approach 2: Viewing Your Modifications to a Shared Folder](#).

Approach 1: Deploying 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 that is shipped with Oracle BI EE contains an example folder structure to help you get started.

 **Note:**

This approach automatically deploys your custom skin to all nodes in your Oracle BI EE 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 does not 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 [Customizing 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. For additional information about updating styles in a dashboard, see Changing the Properties of a Dashboard and its Pages in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

If these entries are not present in the instanceconfig.xml file, then fusionFX is the default style. 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>Skyros</DefaultStyle>
    <DefaultSkin>Skyros</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 `Skyros` above with the word `Alta`.

 **Note:**

These names must match the names given to the folders containing the style and skin. Do not include underscores. For example: If your folder begins with the characters `s_`, such as `s_Skyros`, then omit `s_`.

- c. Save your changes and close the file.

6. Restart Presentation Services. See [About Managing Oracle Business Intelligence Processes](#).

To redeploy the bicustom.ear file, see [Approach 1: Redeploying the "bicustom.ear" File](#).

Approach 1: Redeploying 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 are changing component values, such as an image or font color in your customized files, you do not 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: Deploying 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 in Oracle BI EE as soon as possible.

This approach is also used when your customizations are outside the Presentation Services firewall.

In Oracle BI EE, 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.
 - a. Extract the bicustom.war file from the bicustom-template.ear file.
 - b. Extract the contents of the bicustom.war file to a location accessible to Oracle Weblogic Server (for example, `c:\custom`).
 - c. Edit the files to create your custom style and save your changes. See [Customizing Your Style](#) for additional information.
2. Deploy the custom folder.
 - 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 your custom style (for example, `c:\custom`).
 - f. Click **Next**.
 - g. Select **Install this deployment as an application**.
 - h. 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 from Oracle BI Server, the core server behind Oracle Business Intelligence.

- 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>Skyros</DefaultStyle>
    <DefaultSkin>Skyros</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 *Skyros* above with the word *Alta*.

 **Note:**

The CustomerResourceVirtualPath points to the location entered in the Context Root box.

If you are 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>Skyros</DefaultStyle>
    <DefaultSkin>Skyros</DefaultSkin>
    <SkinMappings>
      <skinMapping>
        <biadfSkinFamily>fusion</biadfSkinFamily>
        <biSkin>FusionFx</biSkin>
      </skinMapping>
    </SkinMappings>
  </UI>
</ServerInstance>
```

```

        <biadfSkinFamily>blafplus-rich</biadfSkinFamily>
        <biSkin>blafp</biSkin>
    </skinMapping>
    <skinMapping>
        <biadfSkinFamily>skyros</biadfSkinFamily>
        <biSkin>Skyros</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: Viewing 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. In Oracle BI EE, 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. Note that this process is dependent on the browser being used.
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.

Customizing 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.

Folder or File Name	Description
filemap.xml ²	<p>The filemap.xml file enables you to specify that your style or skin extends another style or skin. By default, s_Custom¹ extends s_Skyros, which is defined in the filemap.xml file. This means that if a file cannot 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:</p> <pre><FileMap> <Styles Default="s_blafp"> <Hierarchy>s_Skyros/s_Custom</Hierarchy> </Styles> <Skins Default="sk_blafp"> <Hierarchy>sk_Skyros/sk_Custom</Hierarchy> </Skins> </FileMap></pre>
s_Custom	<p>s_Custom is a folder containing the files that define a style named Custom. The prefix, s_, indicates that the folder contains a <i>style</i>. This name displays in the Dashboard Properties dialog, Styles option. For additional information about updating styles in a dashboard, see Changing the Properties of a Dashboard and its Pages in <i>User's Guide for Oracle Business Intelligence Enterprise Edition</i>.</p> <p>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.</p> <p>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.</p> <p>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.</p>

Folder or File Name	Description
sk_Custom	<p>sk_Custom¹ 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 is not provided as part of the bicustom-template.ear file. The prefix, sk_, indicates that the folder contains a <i>skin</i>.</p> <p>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.</p> <p>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.</p> <p>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.</p>
master	<p>The master folder contains all of the files that you likely need to create a custom style.</p>
master.css	<p>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 Oracle BI Publisher. Changing the styles defined in the classes contained in this file widely impacts Oracle BI EE.</p> <p>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.</p>
custom.css	<p>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 upgrades.</p>
graph.xml	<p>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.</p>

Folder or File Name	Description
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 are extending is applicable only for Presentation Services.

Example of Modifying the Skyros Master Branding Class

You can modify the CSS for Skyros 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 [Customizing the Oracle BI Web User Interface](#) for additional information.

Embedding External Content in Dashboards

For security reasons, you can no longer embed content from external domains in dashboards. To embed external content in dashboards, you must edit the `instanceconfig.xml` file.

1. Open the `instanceconfig.xml` file for editing, located in:


```
ORACLE_HOME/user_projects/domains/bi/config/fmwconfig/
biconfig/OBIPS
```
2. Depending on whether you want to embed content from all external domains or a specific external domain, perform one of the following actions:
 - a. To embed content from all external domains, include the elements and their ancestor elements, as shown in the following example:

```
<ServerInstance>
  <Security>
    <ContentSecurityPolicy>
      <PolicyDirectives>
        <Directive>
          <Name>frame-src</Name>
          <Value>*</Value></Directive>
        <Directive><Name>img-src</Name>
```

```

    <Value>*</Value>
  </Directive>
</PolicyDirectives>
</ContentSecurityPolicy>

```

 **Note:**

The wildcard character "*" in the <Value> element enables you to embed content in dashboards from both internal and external domains.

- b. To embed content from a number of specific domains, for example `www.xxx.com`, `www.yyy.com`, and so on, include the name of each domain separated by a space, in the elements and their ancestor elements, as shown in the following example:

```

<ServerInstance>
  <Security>
    <ContentSecurityPolicy>
      <PolicyDirectives>
        <Directive>
          <Name>frame-src</Name>
          <Value>www.xxx.com www.yyy.com</Value></Directive>
          <Directive><Name>img-src</Name>
          <Value>www.xxx.com www.yyy.com</Value>
        </Directive>
      </PolicyDirectives>
    </ContentSecurityPolicy>

```

Keep the following points in mind when you configure the `instanceconfig.xml` file for specific domains:

- The default setting of the `instanceconfig.xml` file enables you to embed content from internal domains only. When you edit the `instanceconfig.xml` file to specify external domains, you overwrite the default setting of the file, thus preventing you from accessing internal domains. To enable you to continue accessing internal domains, you must precede the names of the external domains with the word 'self' followed by a space. For example, to continue to embed content from internal domains and a specific domain (`www.xxx.com`), you enter the following text in the <Value> element:


```
'self' www.xxx.com
```
 - When you configure the `instanceconfig.xml` file for specific domains, you can use the wildcard character to embed content from all sub-domains under the parent domain. For example, when you specify `www.xxx.*.com`, you can embed content from all sub-domains such as `xxx.hr.com`, `xxx.fin.com`, and so on.
3. Save your changes and close the file.

19

Configuring and Managing Agents

This chapter describes how to configure and manage agents in Oracle Business Intelligence.

If your organization licenses Oracle BI Delivers and if you have 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 *Delivering Content* in *User's Guide for Oracle Business Intelligence Enterprise Edition*.

This chapter includes the following sections:

- [How Are Agents Used?](#)
- [How Do Antivirus Software and Privileges Affect Agents?](#)
- [Configuring Settings that Affect Agents](#)
- [Managing Device Types for Agents](#)
- [Monitoring Active Agent Sessions](#)

Note:

If you are migrating an Oracle Business Intelligence environment to a new system, then ensure that you also migrate the Oracle Business Intelligence repository file, the Oracle BI Presentation Catalog, and the Oracle BI Scheduler tables. The Oracle BI Scheduler tables are required for agents.

See [Diagnosing 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, Oracle Business Intelligence users (with the Create Agent privilege) define the operations that the agent is to perform. Oracle BI Server packages information such as priority, delivery devices, and user, into a job, and tells Oracle BI Scheduler when to execute the job. See *What is Oracle BI Scheduler?* in *Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*.

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 Oracle BI 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 Oracle BI Presentation Catalog. See Managing Presentation Services Privileges in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

Note:

If the Oracle BI 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 is set up for authentication in the Oracle BI Administration Tool uses a connection pool with pass-through login. That connection pool cannot be used for any other initialization block or request.

For information about user authentication options, see *Security Guide for Oracle Business Intelligence Enterprise Edition*. For information about pass-through login, see *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Configuring Settings that Affect Agents

You configure settings for agents by changing values for Oracle BI 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 Configuring Presentation Services Settings that Affect Agents](#)
- [Manually Changing Additional Scheduler Settings that Affect Agents](#)
- [What Additional Scheduler Configuration Settings Affect Agents?](#)
- [Controlling Delivery Options for Agents](#)

Manually Configuring 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.

Element	Description	Default Value
Enabled	Specifies whether Oracle BI 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

Element	Description	Default Value
DefaultDeliveryFormat	<p>Specifies the default format for sending emailed reports through an agent.</p> <p>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:</p> <p style="margin-left: 40px;">html pdf excel text</p>	html

Manually Changing 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 instanceconfig.xml file to change these settings. You must apply changes to both the primary and secondary scheduler's instanceconfig.xml in a cluster.

1. Open the Oracle BI Scheduler version of the instanceconfig.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_Dir is shown in the following example:

```
<xs:element name="Log_Dir" type="xs:string" default="ibots" minOccurs="0">
  <xs:annotation>
    <xs:documentation xml:lang="en">
      The directory where Agent logs are stored.
    </xs:documentation>
  </xs:annotation>
</xs:element>
```

Note:

You cannot specify values for user name and password in the instanceconfig.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 Oracle Business Intelligence.

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

Element	Description	Default Value
TABLE_JOBS	<p>Specifies the name of a database table used to store information about scheduled jobs.</p> <p>For information about modifying the database table names, see <i>Changing Oracle BI Scheduler Table Names in Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition</i> and see <i>Oracle Business Intelligence Applications Installation and Configuration Guide</i>.</p>	S_NQ_JOB
TABLE_INSTANCES	<p>Specifies the name of a database table used to store information about job instances.</p>	S_NQ_INSTANCE
TABLE_PARAMS	<p>Specifies the name of a database table used to store information about job parameters.</p>	S_NQ_JOB_PARAM
TABLE_ERRMSG	<p>Specifies the name of a database table used to store information about job instances that do not complete successfully.</p>	S_NQ_ERR_MSG
SchedulerScriptPath	<p>Refers to the path where Oracle BI Scheduler-created job scripts are stored. In general, do not add or remove scripts from this directory. By default, this field is set to:</p> <p>BI_DOMAIN/servers/ obisch1</p>	scripts\scheduler
DefaultScriptPath	<p>Specifies the path where user-created job scripts (not agents) are stored.</p> <p>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 is not examined. By default, this field is set to:</p> <p>BI_DOMAIN/servers/ obisch1</p>	scripts\common

Element	Description	Default Value
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 is not 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. Do not 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
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 does not 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 executes jobs at runtime.	1

Element	Description	Default Value
MaxExecThreads	Specifies the maximum number of multiple threads in the Oracle BI Scheduler thread pool that executes jobs at runtime.	100
PauseOnStartup	Specifies that no jobs are executed when Oracle BI Scheduler starts. While Oracle BI Scheduler pauses, users can add, modify, and remove jobs. However, no jobs execute. 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 DN's.	No 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

Element	Description	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
SmtxCipherList	Specifies the list of ciphers that match the cipher suite name that the SMTP server supports. For example, RSA+RC4+SHA. See <i>Enabling SSL in a Configuration Template Configured System</i> in <i>Security Guide for Oracle Business Intelligence Enterprise Edition</i> .	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 Diagnosing 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 executed, 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 Business Intelligence 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 <i>BI_DOMAIN/servers/obisch1/logs</i> A new log file named Agent-<Job number>-<Instance number>.log is created for each job instance. The Job Manager can also be used to override the Debug setting for an individual job. See Diagnosing 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: <i>BI_DOMAIN/bi/servers/obisch1/logs</i>	true

Controlling 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 *Setting Up the SA System Subject Area in Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition* for information.

Managing Device Types for Agents

You can use different device types from such categories as mobile phones and pagers to deliver the content of agents to users. You can create, view, edit, and delete device types for a device category. Many device types are provided automatically. You can add types that are required for your users.

Note:

You can only view system-seeded device types (such as AT&T Wireless); you cannot edit or delete them.

The capability to manage device types is available to users who have the Manage Device Types privilege. See *Managing Presentation Services Privileges Using Application Roles in Security Guide for Oracle Business Intelligence Enterprise Edition*.

1. Log in to Oracle Business Intelligence.
2. In the global header, click **Administration**.
3. Click the **Manage Device Types** link to display the Manage Device Types page.
4. Click the **Create New Device Type** link.
5. Complete the Create New Device Type dialog, and click **OK**.
6. Click **Create Device Type** to return to the Manage Device Types screen.

To view or edit a device type:

1. In the global header, click **Administration**.
2. Click the **Manage Device Types** link.
3. Click the **Edit** button for the appropriate device type.
4. Complete the Edit Device Type dialog, and click **OK**.

To delete a device type:

1. In the global header, click **Administration**.
2. Click the **Manage Device Types** link.
3. Click the **Delete** button for the device type to delete.
A confirmation box is displayed.
4. Click **OK** to confirm the deletion.

Monitoring Active Agent Sessions

Using the Manage Agent Sessions page in Oracle BI 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.

Configuring Advanced Options for Mapping and Spatial Information

This chapter describes advanced configuration options that you can set for map views in Oracle Business Intelligence.

This chapter includes the following sections:

- [Configuring MapViewer to Support Map Views](#)
- [Manually Configuring for Map Views](#)
- [Inserting Text on a Map](#)
- [Configuring Maps for External Consumption](#)

See [Configuring Mapping and Spatial Information](#). Before configuring for map views, ensure that you are familiar with the information in the following guides:

- *Map Visualization Developer's Guide*, which is part of the Oracle Fusion Middleware documentation library.
- *Oracle Spatial Developer's Guide*, which is part of the documentation library for Oracle Database.

Configuring MapViewer to Support Map Views

Oracle Fusion Middleware MapViewer is installed as part of Oracle BI Enterprise Edition and deployed in the same domain as Oracle BI EE 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. See [Configuring MapViewer](#) in *Map Visualization Developer's Guide*.

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 is not a major concern, then you can use a MapViewer instance that is co-located with Oracle BI EE 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 same domain as Oracle BI EE 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 cannot communicate with the remote instance for resources. Browsers do not 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 Configuring for Map Views](#).
- Edit the proxy_enabled_hosts element in the mapViewConfig.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>
```

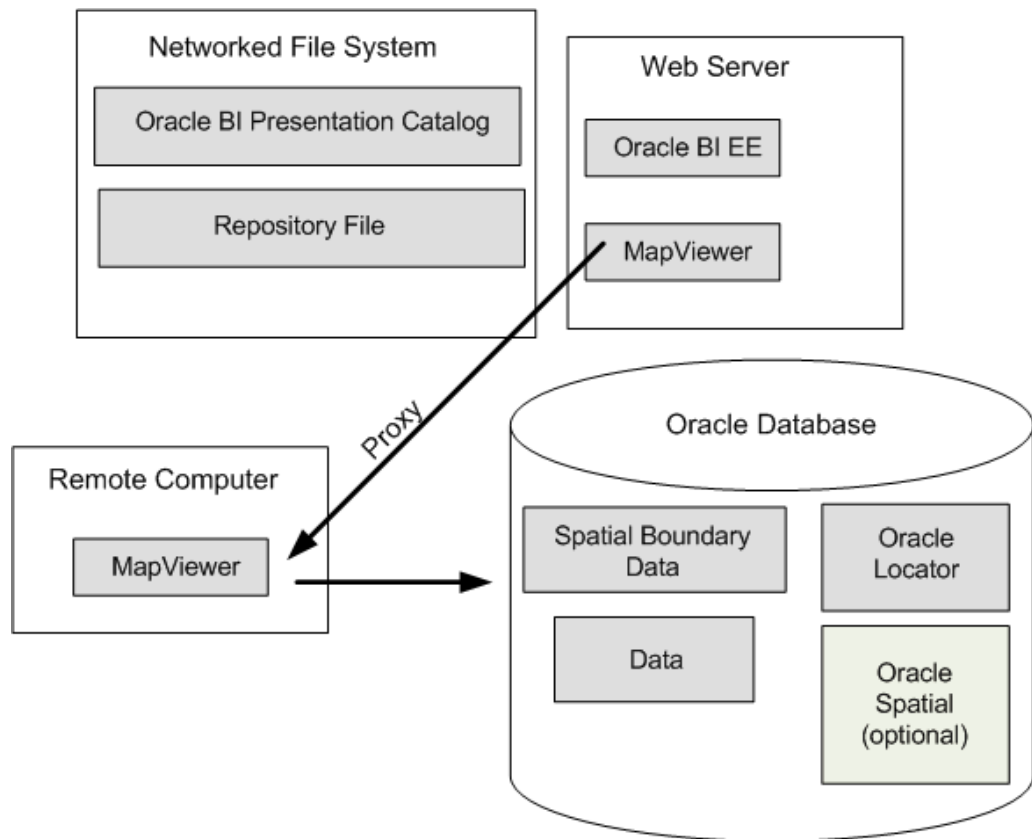
The mapViewConfig.xml file is located in the following directory:

```
ORACLE_BI1\bifoundation\jee\mapviewer.ear\web.war\WEB_INF  
\conf
```

 **Note:**

If you edit the mapViewConfig.xml file, then your edits are overwritten when you upgrade or patch Oracle Business Intelligence. Before upgrading or patching, ensure that you make a backup copy of the mapViewConfig.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 Business Intelligence.

The figure below shows the preferred architecture for map views, which provides better performance through a proxy than the default architecture that is shown in the table below. You can store the data either in an Oracle Database or in other databases that Oracle BI EE supports.



Manually Configuring for 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:
 - ColocatedOracleMapViewContextPath — Specifies the context path of the MapViewer instance that is co-located with Oracle BI EE. The default value is /mapviewer.
 - RemoteOracleMapViewAbsoluteURL — 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:

```
<RemoteOracleMapViewAbsoluteURL>http://remoteserver:9704/mapviewer </
RemoteOracleMapViewAbsoluteURL>
```

- 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.

- `SyndicatedOracleMapViewContextPath` — Specifies the URL of the MapViewer instance for embedding maps in external pages. For details and an example, see [Configuring Maps for External Consumption](#).
3. Include the elements and their ancestor elements as appropriate, as shown in the following example.

```
<ServerInstance>
  <SpatialMaps>
    <ColocatedOracleMapViewContextPath>/mapviewer</ColocatedOracleMapViewContextPath>
    <RemoteOracleMapViewAbsoluteURL></RemoteOracleMapViewAbsoluteURL>
    <LayerDataLayout>
      <MaxRecords>600</MaxRecords>
    </LayerDataLayout>
  </SpatialMaps>
</ServerInstance>
```

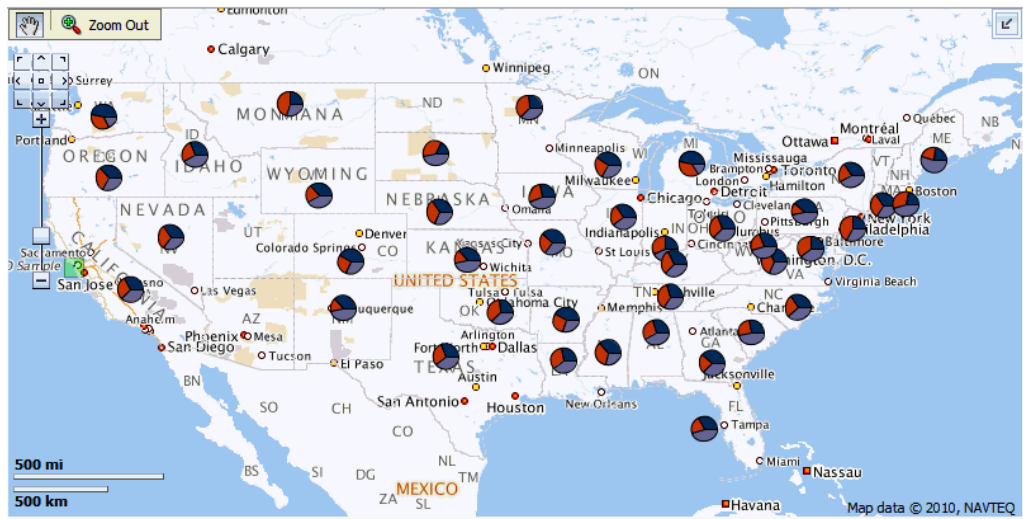
4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Inserting 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 BI EE 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.
See *Map Visualization Developer's Guide*.
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 © 2010, NAVTEQ</copyright>
```

4. Click **Submit** to save your changes.
5. If you do not see the updated text on the map, then click the browser's **Refresh** button to refresh the map.

Configuring 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 BI EE. In this case, set the **SyndicatedOracleMapViewerContextPath** element. When the server for Oracle BI 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>
```

21

Configuring 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 Oracle BI 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 Oracle Business Intelligence.

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: <i>protocol://server/virtualpath</i> where <i>virtualpath</i> 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	<p>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 <code>ORACLE_HOME\bi\bifoundation\web\appv2\res</code>.</p> <p>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.</p>	Varies
CustomerResourceVirtualPath	<p>Specifies the virtual path used for resource files that are not part of a default installation as specified in the <code>CustomerResourcePhysicalPath</code> element.</p>	No default value
ForceAbsoluteResourceURL	<p>Specifies whether Presentation Services always generates fully qualified URLs for resource files that have fully qualified virtual paths.</p> <p>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.</p>	false
ResourcePhysicalPath	<p>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 <code>ORACLE_HOME\bi\bifoundation\web\appv2\res</code>.</p> <p>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.</p> <p>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 <code>ResourceVirtualPath</code> element.</p>	No default value

Element	Description	Default Value
ResourceServerPrefix	<p>Specifies how Presentation Services generates URLs for static resources such as images, script files, style sheets, and other user-specified files. The default is <i>protocol://server</i> from the AbsoluteCommandURLPrefix element.</p> <p>If you explicitly specify a value, then it must be of this form:</p> <pre>protocol://server</pre> <p>If you specify a virtual path, then it is removed.</p> <p>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.</p>	<p><i>protocol://server</i></p>
ResourceVirtualPath	<p>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.</p> <p>For generating relative URLs, the virtual path defaults to <i>res</i>, if the resource folder is present under the same virtual directory as the Oracle BI Presentation Services DLL files.</p> <p>For generating absolute URLs, the value of the AbsoluteCommandURLPrefix element is used as the default.</p> <p>The value must be a fully qualified virtual path of this form:</p> <pre>'/VirtualPath'</pre> <p>If you omit the leading slash, then one is added.</p>	<p><i>res</i></p>

Part VII

Managing the Life Cycle

This part describes life cycle management tasks for Oracle BI Enterprise Edition. Life cycle management consists of installing, upgrading, 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 Business Intelligence*.

For information about upgrade, see *Upgrading Oracle Business Intelligence*.

This part includes the following chapters on life-cycle management tasks:

- [Patching Oracle Business Intelligence Systems](#)
- [Moving Oracle Business Intelligence Between Environments](#)
- [Backup and Recovery of Oracle Business Intelligence Systems](#)

Patching Oracle Business Intelligence Systems

This chapter provides information on patching Oracle Business Intelligence. See *Oracle Fusion Middleware Patching Guide*.

This chapter includes the following sections:

- [About Patching Oracle Business Intelligence Systems](#)
- [What Is Patched for the Oracle Business Intelligence Platform?](#)
- [Rolling Back a Platform Patch](#)
- [Determining Current Patch Levels](#)

About Patching Oracle Business Intelligence Systems

Patching involves copying a small collection of files over an existing installation.

A patch is normally associated with a particular version of an Oracle product and involves updating from one minor version of the product to a newer minor version of the same product (for example, from version 12.2.1.1.0 to version 12.2.1.2.0). 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 BI EE system that is 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 BI EE 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 12c BI Server Installation

This section explains how you update an existing 12.2.x BI Server installation.

Assumptions:

- For one-off patches you download `<patch-number>.ZIP` from:

<https://support.oracle.com>

- For all other updates you download `bi_platform-12.2.x.y.z-<platform>.<extension>` from:

<https://otn.oracle.com>

- Software updates will only change `ORACLE_HOME` and `PRODUCT_HOME` content, `DOMAIN_HOME` and `BI_DATA_HOME` will be untouched.

Pre-conditions:

All product processes have been shutdown.

1. For each distinct `ORACLE_HOME`, run the Oracle Universal Installer.

For information, see new server install in *Installing and Configuring Oracle Business Intelligence*.

2. (Optional) For each distinct `ORACLE_HOME`, you launch `OPatch` (`ORACLE_HOME/OPatch/opatch`) with the zip file.

This will update 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 Business Intelligence Platform?

Oracle Business Intelligence platform patching applies patches for binary files with extensions such as `DLL`, `JAR`, and `EXE`.

Oracle Business Intelligence platform patching does not 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

Nondesign-time metadata that is stored in database schemas (including schemas for the Scheduler, usage statistics, event polling, repository files, and the Oracle BI Presentation Catalog) is not patched.

Other platform metadata (such as repository files and Oracle BI Presentation Catalog files) that are delivered in the context of an application are patched, but as part of an applications patch and not as part of a platform patch.

Rolling 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 BI EE 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 [Determining Current Patch Levels](#).

Determining Current Patch Levels

Each Oracle home must be patched to the same version as OPatch to ensure that Oracle BI EE 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*.

Moving Oracle Business Intelligence Between Environments

This chapter describes how to move Oracle Business Intelligence between environments. You can move Oracle Business Intelligence from a test to a production environment or to a new environment, as follows:

- [Moving Between Environments](#)
- [Moving to a New Environment](#)
- [Upgrading from 11g to 12c](#)
- [Migrating the Whole Server](#)

Moving Between Environments

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.

Assumptions

- You must have file system (offline) permissions.
- Commands can be run offline only.
- Command is read-only for export (source system is unchanged).

Prerequisites

- Uses WLST scripting.
See [Using the WebLogic Scripting Tool \(WLST\)](#).
- BI deployments A and B service instances exist in different domains.
- Changes will have been made to BI deployment A service instance.
- Admintool (datamodel), Answers (catalog), Fusion Middleware Control (security), or WLST. These tools are only required if you want to make changes to metadata.
- Command is read-only.

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 datasources 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 upgraded metadata from one system to another after it has been tested (no users in common, different datasource security).

To support these options the export command has optional parameters to export user folder content, or connection pool credentials as follows:

```
exportServiceInstance(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 `exportServiceInstance` command to export the TEST service instance to a BAR file (with or without User folders and connection credentials).

For example:

```
exportServiceInstance('/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 service instance with key `ssi`, to `/u01/exportDir/ssi.bar`.

3. Run the `importServiceInstance` command to import the exported TEST BAR file into the PROD service instance.

```
importServiceInstance('/u01/Oracle/Middleware/Oracle_Home/user_projects/domains/  
bi', 'ssi', '/u01/exportDir/ssi.bar')
```

This example imports the 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:

- PROD Service Instance metadata will be replaced with TEST metadata.
- To restate, any content created directly in PROD Service Instance will be replaced or lost.
- No configuration is changed on domain hosting PROD Service Instance
- No metadata or configuration is changed on domain hosting TEST Service Instance.

Moving to a New Environment

You can move (or migrate) Oracle Business Intelligence to a new environment by re-creating an existing Oracle BI 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.

For information about migrating to a new environment, see *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Upgrading from 11g to 12c

You can upgrade an 11g deployment by migrating 11g metadata to a new 12c deployment.

The process is an out-of-place upgrade that requires a 12c domain to be created as the target. You first create an export bundle from a read-only 11g certified release, and then use it to reconfigure a new 12c domain.

For information about migrating and upgrading, see *Upgrading Oracle Business Intelligence*.

Migrating the Whole Server

Oracle Business Intelligence supports whole server migration, in which a WebLogic Server instance is migrated to a different physical computer upon failure, either manually or automatically.

See Whole Server Migration in *Administering Clusters for Oracle WebLogic Server* and Configuring Server Migration for an Enterprise Deployment in *Enterprise Deployment Guide for Oracle Business Intelligence*.

Backup and Recovery of Oracle Business Intelligence Systems

This chapter describes general information on backup and recovery for Oracle Business Intelligence. 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.

Backup and recovery for Oracle Business Intelligence is described in Backup and Recovery Recommendations for Oracle Business Intelligence in *Administering Oracle Fusion Middleware*.

Disaster recovery for Oracle Business Intelligence is described in Recommendations for Oracle Fusion Middleware Components in *Oracle Fusion Middleware Disaster Recovery Guide 12c R2 (12.2.1)*.

For more information about backing up and restoring Oracle Business Intelligence metadata, see [Managing Metadata and Working with Service Instances](#).

Part VIII

Using Oracle Essbase With Oracle Business Intelligence

Oracle Essbase works with Oracle Business Intelligence to create a complete set of data analysis and presentation features.

This part provides information for using Oracle Essbase when it is installed with Oracle Business Intelligence. It includes the following chapters:

- [Introduction to Using Oracle Essbase in Oracle Business Intelligence](#)

Introduction to Using Oracle Essbase in Oracle Business Intelligence

This chapter explains what you need to know about using Oracle Essbase when installed with Oracle Business Intelligence. The chapter includes a high-level roadmap; contains information about installation, system administration, service-level maintenance and security, and links to alternative sources of information.

This chapter includes the following sections:

- [Overview](#)
- [High-Level Roadmap for Working with Essbase When Installed with Oracle Business Intelligence](#)
- [Performing Tasks When Essbase Is Installed with Oracle BI EE Compared to Performing the Same Tasks in EPM and Information on Which Guides to Consult](#)
- [Installing Essbase with Oracle Business Intelligence](#)
- [Configuring Security for Essbase in Oracle Business Intelligence](#)
- [Managing Essbase System Administration in Oracle Business Intelligence](#)
- [Working with Essbase Data in Oracle Business Intelligence](#)
- [Where Can I Learn More Information About Essbase?](#)

Overview

You can easily deploy Essbase and associated tools when installing Oracle Business Intelligence and benefit from using shared administration, user management, installation, and configuration support functionality that is also available to Oracle Business Intelligence users.

When you deploy Essbase with Oracle Business Intelligence, you can access Essbase multidimensional databases that provide multidimensional analysis, enabling rapid development of custom analytic applications. Essbase enables you to develop and manage analytic applications that model complex scenarios, forecast business trends, and perform "what-if" analyses. Essbase supports fast query response times using pre-aggregated dimensions, for large numbers of users, for large data sets, and for complex business models. Essbase can be configured to use with a range of data sources.

Essbase Studio and EAS can be deployed by the 11g EPM installer in a separate domain, and can be used with the 12c Essbase server installed through Oracle Business Intelligence.

This chapter includes information about installing, configuring, managing and securing Essbase installed through Oracle Business Intelligence as well as creating, securing, and accessing Essbase databases using Oracle Business Intelligence.

This chapter does not describe using Essbase when installed with the EPM System Installer. For information about using Essbase and associated tools when installed

with the EPM System Installer, see the EPM System deployment documentation, located on the **Deployment and Installation** tab in the EPM System Documentation Library.

For information on which guide to refer to for information about Essbase and associated tools, when installed with Oracle Business Intelligence, see [Performing Tasks When Essbase Is Installed with Oracle BI EE Compared to Performing the Same Tasks in EPM and Information on Which Guides to Consult](#).

Any other use of Essbase in the context of being used as a data source for Oracle Business Intelligence should leverage an EPM deployment of Essbase and related tools from 11g (see the EPM deployment guides). In future releases the role of Essbase deployed through Oracle Business Intelligence will expand. Therefore Oracle recommends installing the Essbase software to avoid future challenges with expanding the BI Domain. See Understanding Essbase Deployed in BI 12.2.1 in *Oracle Essbase Database Administrator's Guide*.

High-Level Roadmap for Working with Essbase When Installed with Oracle Business Intelligence

Read this section before you start working with Essbase when installed with Oracle Business Intelligence.

This high-level roadmap explains the tasks to perform and what choices are available. This section also indicates when to refer to the Oracle Business Intelligence documentation or the EPM System documentation.

1. Read the rest of this chapter for an overview of Essbase concepts and configuration tasks.

For example, see [Performing Tasks When Essbase Is Installed with Oracle BI EE Compared to Performing the Same Tasks in EPM and Information on Which Guides to Consult](#).

2. Select the Essbase Suite option to install Essbase with Oracle Business Intelligence as described in [Installing Essbase with Oracle Business Intelligence](#).
3. Perform essential security tasks as described in [Configuring Security for Essbase in Oracle Business Intelligence](#).

Essbase uses the Oracle Fusion Middleware security model, which Oracle Business Intelligence also uses. For more information, see *Security Guide for Oracle Business Intelligence Enterprise Edition*.

4. Perform essential system administration tasks as described in [Managing Essbase System Administration in Oracle Business Intelligence](#).

Perform these tasks after installing Essbase with Oracle Business Intelligence to ensure that Essbase is correctly configured.

5. Create, manage and use Essbase cubes as described in [Working with Essbase Data in Oracle Business Intelligence](#).

Performing Tasks When Essbase Is Installed with Oracle BI EE Compared to Performing the Same Tasks in EPM and Information on Which Guides to Consult

When you install Essbase with Oracle Business Intelligence, you do not perform a full Enterprise Performance Management System installation. Instructions for completing certain tasks can vary from those documented in the EPM documentation.

Use the table below to help you decide what to use to perform certain tasks in Oracle Business Intelligence compared to EPM systems, and which guides to reference.

Essbase-Related Tasks Performed in Oracle BI and EPM Systems	How to Perform Essbase-Related Tasks in an Oracle BI System and Which Guides to Reference	How to Perform Essbase-Related Tasks in an 11g EPM System and Which Guides to Reference
Authentication	Essbase should be setup as part of the installation process and all security in the 12.2.1 release should be handled automatically through the BI security. Refer to EPM 11g guides.	When using an 11g EPM system, use HSS, MaxL. Refer to EPM guides.
Authorization	Essbase should be setup as part of the installation process and all security in the 12.2.1 release should be handled automatically through the BI security. Refer to EPM 11g guides.	When using an 11g EPM system, use HSS, MaxL. Refer to EPM guides.
Data Security (Provisioning)	When using Essbase and associated tools installed with Oracle Business Intelligence, use Fusion Middleware Control and Security. Refer to EPM 11g guides. There is no need to define Essbase filters for row level security; it is handled by BI security privileges. Filters will need to be created for Essbase when there is a 12c cube that will be accessed directly by the user, non-acceleration cubes.	When using an 11g EPM system, use HSS. Refer to EPM guides.
Data Security (filter definitions)	When using Essbase installed with Oracle Business Intelligence, use MaxL. Refer to EPM 11g guides. There is no need to define Essbase filters for row level security; it is handled by BI security privileges. Filters will need to be created for Essbase when there is a 12c cube that will be accessed directly by the user, non-acceleration cubes.	When using an 11g EPM system, use EAS (GUI) or Maxl (Command Line). Refer to EPM guides.

Essbase-Related Tasks Performed in Oracle BI and EPM Systems	How to Perform Essbase-Related Tasks in an Oracle BI System and Which Guides to Reference	How to Perform Essbase-Related Tasks in an 11g EPM System and Which Guides to Reference
System Configuration	When using Essbase and associated tools installed with Oracle Business Intelligence, use Fusion Middleware Control, and essbase.cfg. Refer to Oracle Business Intelligence guides.	When using an 11g EPM system, use a text editor and MaxL, essbase.cfg. Refer to EPM guides for EAS and Studio.
Backup and Recovery	When using Essbase and associated tools installed with Oracle Business Intelligence use Fusion Middleware Control. Refer to <i>Administering Oracle Fusion Middleware</i> and Oracle Business Intelligence guides.	Refer to EPM guides, see also for backing up Studio.
Migration (Test to Production)	Migration should use a file system copy for artifacts while partitions, CDF/M, security filters and data should be exported via MaxL for EPM column.	Refer to Lifecycle Management guides.
Essbase cubes	When using Essbase you use the Acceleration Wizard.	Build and manage cubes using EAS and Studio. Refer to EPM guides (references to EAS and Studio). Refer to <i>Oracle Essbase Database Administrator's Guide</i> .
Capacity	N/A, Uses Active/Passive model (one cluster only). Refer to Oracle Business Intelligence guides.	Use N Clusters 1, 2, 3, 4 agents. Uses the Active/Active or Active/Passive model. Refer to EPM guides.
Availability	Use Fusion Middleware Control. Uses - Active/Passive model. Refer to Oracle Business Intelligence guides.	Refer to EPM guides. Uses Active/Passive model.
Financial Reporting	Financial Reports, Calculation Manager and Workspace are not part of the 12c deployment.	Refer to EPM guides.
Calculation Manager	Financial Reports, Calculation Manager and Workspace are not part of the 12c deployment.	Refer to EPM guides.
Workspace	Financial Reports, Calculation Manager and Workspace are not part of the 12c deployment.	Refer to EPM guides.
APIs	No need for custom API work in 12.2.1.	NA

Installing Essbase with Oracle Business Intelligence

You can install Essbase when you install Oracle Business Intelligence.

When you install Essbase with Oracle Business Intelligence, Essbase databases become available to Oracle Business Intelligence users, and you can manage Essbase components using a combination of Oracle Business Intelligence tools and

Essbase EPM System tools. For information about which version of Essbase is installed, go to the Certifications page in Oracle Support and check the product details for appropriate releases of Oracle Business Intelligence Enterprise Edition with Oracle Essbase at:

<https://support.oracle.com>

 **Note:**

You cannot add Essbase to an existing Oracle Business Intelligence installation although you can still use Essbase as a data source. The only way to install Essbase with Oracle Business Intelligence is to perform a new Oracle Business Intelligence installation and select the Essbase component.

You install EAS and Studio using the EPM installation. See the EPM guides for more details.

This topic contains the following sections:

- [Installing Essbase](#)
- [Selecting the Essbase Suite Option During Install](#)
- [Limitations for Using Client Tools when Essbase Is Installed with Oracle Business Intelligence](#)
- [Essbase Features Not Supported when Essbase Is Installed with Oracle Business Intelligence](#)

Installing Essbase

You install Essbase Suite as part of an Oracle Business Intelligence Enterprise installation.

See *Installing and Configuring Oracle Business Intelligence*.

The Enterprise Install type creates and configures a single WebLogic Server Domain that contains a cluster with an Administration Server, and a Managed Server. If you select Essbase Suite during the install, then Essbase JEE components and Oracle Business Intelligence JEE components are installed on Managed Servers. See [Selecting the Essbase Suite Option During Install](#).

Selecting the Essbase Suite Option During Install

You can install JEE applications and services during install.

Selecting the Essbase Suite option during the install does the following:

- Pre-selects the **Oracle Business Intelligence Enterprise Edition** option.
- Installs the following Essbase-related JEE applications and services into the local Oracle BI EE Managed Server in an Enterprise Install:
 - Essbase Agent
 - Cube Deployment Services

- Hyperion Provider Services (APS)
- Installs the following Essbase server processes:
 - Essbase App.
- Enables you to download the following Essbase-related client applications from the Download BI Desktop Tools option, which is available in the Get Started area of the Home page for Oracle BI EE:
 - Smart View for Office (EPM application)

See Downloading BI Desktop Tools in *User's Guide for Oracle Business Intelligence Enterprise Edition*.
- Enables users to log in to the following Essbase command line tools:
 - ESSCMD
 - ESSMSH (MaxL)
- Enables users to log in through the following supported Essbase API languages:
 - JAPI
 - C-API

 **Note:**

Client applications and API languages that are not included in this section are not supported with Essbase and related components installed with Oracle Business Intelligence.

Limitations for Using Client Tools when Essbase Is Installed with Oracle Business Intelligence

There are limitations for using the following Essbase tools when Essbase is installed with Oracle Business Intelligence.

- MaxL command line interface

The MaxL command line supports most of the tasks for Oracle Business Intelligence Essbase that are also possible with Oracle EPM System Essbase.

 **Note:**

You cannot use the MaxL command line to provision security.

Essbase Features Not Supported when Essbase Is Installed with Oracle Business Intelligence

When Essbase is installed with Oracle Business Intelligence, you do not get a full Enterprise Performance Management installation.

The following Essbase features are not supported:

- Permission grants are not supported at the database level. For example, if you have two multidimensional databases that are used by an application, then permission grants apply equally to the multidimensional databases in an application.
- Multiple clusters are not supported.
- The Active-Active failover process is not supported. You can have only one machine or cluster active at a time, and you cannot load balance across computers or clusters.
- The audit report showing what users, groups and application roles can do in Essbase, when Essbase is installed as part of an Enterprise Performance Management System, is not available in this release. However, you can use Oracle Fusion Middleware to view the Essbase permissions assigned to application roles.
- There might be other Enterprise Performance Management features that are not supported when Essbase is installed with Oracle Business Intelligence. To avoid misunderstandings, do not use EPM System or Essbase documentation (when Essbase is installed with Oracle Business Intelligence), unless directed to do so in this documentation.

Configuring Security for Essbase in Oracle Business Intelligence

This section explains typical Essbase-related security configuration when Essbase is installed with Oracle Business Intelligence. You might also refer to this section if you maintain Essbase security in an existing installation of Oracle Business Intelligence.

The Oracle Business Intelligence Installer configures Essbase to use Oracle Fusion Middleware security for authentication and authorization. You can secure an Essbase environment to provide equivalent functionality to Essbase secured through Hyperion Shared Services. Exceptions to this are documented in the Oracle Business Intelligence Certification Matrix.

Essbase installed using the Oracle Business Intelligence installer cannot use Native Essbase or Hyperion Shared Services (HSS) security. However, when you install Essbase with Oracle Business Intelligence, the Common Security Service (CSS) token-based identity assertion continues to be available and enables Oracle Business Intelligence to connect to Essbase data sources (both Essbase installed with Oracle Business Intelligence and Essbase installed with EPM) with the credentials of the end user. For this mechanism to work with an Essbase data source external to the Oracle Business Intelligence installation, you must follow the documentation. Also note that if multiple Essbase data sources are being used by Oracle Business Intelligence and there is a requirement to use this mechanism, all Essbase data sources must use the same shared secret for producing CSS tokens. See *Section Configuring Oracle Business Intelligence to Use Hyperion SSO Tokens when Communicating with Essbase, Hyperion Financial Management, Hyperion Planning, and EPM Workspace*.

The Oracle Business Intelligence installer pre-configures core users, groups, application roles, credentials, permissions, and privileges for Essbase. There is no automated migration of security artifacts from Essbase installed with the EPM System Installer to Essbase installed with Oracle Business Intelligence. You must configure the equivalent authentication and authorization mechanism for Essbase when it is installed with Oracle Business Intelligence.

Topics

- [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#)
- [Enabling Drill-through Reports in Oracle BI EE 12.2.1.x](#)
- [Configuring Data-Level Security Using Essbase Filters](#)
- [Configuring Access to Essbase Calculations](#)
- [Changing Essbase Ports in Oracle Business Intelligence](#)
- [Resource Permissions Reference for Essbase and Associated Tools](#)

Enabling Users to Perform Specific Actions in Essbase and Associated Tools

You enable users to perform specific actions in Essbase and associated tools (for example, read and write, use calculations, use filters, use specific filters) by granting resource permission definitions to application roles.

Appropriate resource permissions must be defined first (see [Configuring Data-Level Security Using Essbase Filters](#) and [Configuring Access to Essbase Calculations](#)). Resource permissions contain definitions of Essbase actions that a user can perform in an Essbase application. Essbase actions are derived from Essbase resource types, which are linked to resource permissions.

Resources are hierarchical; therefore global, cluster, and application level resources are listed.



Note:

The Essbase actions described here are not the same as actions in Oracle Business Intelligence.

The installation process grants default Essbase resource permissions to existing application roles in Oracle Business Intelligence.

For more details about the default Essbase resource permissions configured by default when you install Essbase with Oracle Business Intelligence, see [Resource Permissions Reference for Essbase and Associated Tools](#).



Note:

The BIConsumer application role is granted to all users, and has oracle.essbase.server /EssbaseCluster-1 access and oracle.essbase.application /EssbaseCluster-1 user_filter. This gives all users access to Essbase by default.

 **Note:**

Resource permissions are stored by default in a file-based shared policy store, but you can reassociate them to an OID LDAP shared policy store. See *Using Alternative Authentication Providers in Security Guide for Oracle Business Intelligence Enterprise Edition*.

 **Note:**

Parent roles inherit permission grants through child group or role members. Permission grants are cumulative; for example, a user who has a grant of oracle.essbase (filter) through an application role but not granted through another role, is still considered to have the oracle.essbase (filter) role.

Use the following steps to grant resource permissions to users and application roles.

1. If you are granting resource permission definitions to Essbase filters or Essbase calculations, then the filters or calculations must already exist.

If they do not exist, then follow one of these links:

- [Configuring Resource Permission Definitions for Essbase Filters](#)
- [Configuring Resource Permission Definitions for Essbase Calculations](#)

2. Log in to Oracle Fusion Middleware.
See [Logging into Fusion Middleware Control to Manage Oracle Business Intelligence](#).
3. Select **Business Intelligence**, then **coreapplication**.
4. From the Business Intelligence Instance menu (or right-click **coreapplication**), select **Security** and **Application Policies**.
5. Select the obi Application Stripe.
6. Select a Principal Type and click **Search** to display a list of application roles.

Best practice is to assign permissions to application roles.

This search returns only principals that already have policies or permissions assigned. If you have a new application role that does not have any permissions yet, then you must click **Create** or **Create Like**.

 **Note:**

Non-Essbase permissions might also be displayed for the selected principal.

7. Click **Create** to display the Create Application Grant page.
8. Click **Add** in the Grantee section, to add an application role.
9. Select Application Role from the **Type** list and click the **Search** arrow.

10. Click **OK**.
11. Click **Add** in the Permissions section to display the Add Permission page.
12. Click **Resource Types** and select an appropriate resource type from the list.
For example, select **oracle.essbase.application**.
13. Click the **Search Arrow** button to confirm your selection.

 **Note:**

There is no synchronization between the filters or applications created in EAS and the list returned in the policy store. Therefore, you must manually enter the Resource Name in the following steps the first time that you provision a particular resource (server, application, filter or calculation).

14. Click **Continue** to display the Add Permission page.
15. Enter appropriate information into the **Resource Name** and **Resource Type** fields.

For example, to configure access permissions to use any filters within the Demo application, enter the following information:

Resource Type - `oracle.essbase.application`

Resource Name - `/EssbaseCluster-1/Demo`

Permission Actions - `use_filter`

For more examples of what to enter in the **Resource Type**, **Resource Name**, and **Permission Actions** fields when you grant resource permissions to filters or calculations, see the following:

- [Configuring Resource Permission Definitions for Essbase Filters](#)
- [Configuring Resource Permission Definitions for Essbase Calculations](#)

 **Note:**

You can also manually enter permission actions for the resource into the **Permission Actions** field. Essbase actions are hierarchical, such that actions higher in the list include the lower members. For example, selecting **read** grants **read** and **restart** permissions to the `/EssbaseCluster-1/Demo` Essbase application.

16. Click **Select**.
17. Click **OK**.

Enabling Drill-through Reports in Oracle BI EE 12.2.1.x

Essbase Studio is not provided with Essbase Release 12.2.1.x. To use Essbase Studio to deploy Essbase cubes and enable drill-through functionality with Essbase

12.2.1.x you must install EPM 11g on a separate host and complete some configuration steps.

Use the following steps to run Essbase Studio 11g with Essbase 12.2.1.x.

1. Install EPM 11.1.2.4 and apply the latest Opatch.
2. Install BI 12.2.1 on a different host.
3. Update the Essbase Studio server properties file with `server.trustedAuthenticationHost=<HOSTNAME>:<PORT>`, where HOSTNAME is the host where Essbase is running, and PORT is the Essbase port.

 **Note:**

The server.properties file is located in ORACLE_INSTANCE/EssbaseStudio/essbasestudio1/bin in the EPM 11.1.2.4 environment.

4. Stop and restart Essbase Studio server.
5. Stop the BI instance.
6. Copy cpld.jar from 11.1.2.4 EPM:
`${EPM_ORACLE_HOME}/common/essbase-studio-sdk/11.1.2.0/lib`
 To BI12c:
`${MIDDLEWARE_HOME}/user_projects/domains/${DOMAIN_NAME}/lib`
7. Start Oracle Business Intelligence.

 **Note:**

You cannot deploy and access drill-through reports on multiple BI Essbase instances. Multiple Essbase instances are not supported.

Configuring Data-Level Security Using Essbase Filters

Data-level security enables you to restrict the dimensional data that users see in Essbase multidimensional databases. You secure data-level access for Oracle Business Intelligence users by configuring resource permission definitions on Essbase filters and granting them to application roles.

See [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#).

Topics

- [What Are Essbase Filters?](#)
- [Configuring Resource Permission Definitions for Essbase Filters](#)
- [Securing Data Access with Essbase Filters](#)
- [When Are Filter Access Permission Grant Changes Consumed?](#)
- [How Do Filter Permission Grants Differ Between Oracle Business Intelligence and EPM?](#)

What Are Essbase Filters?

An Essbase filter is an Essbase resource that provides an access control mechanism for dimensional data. For example, a filter might restrict a user to view or update data only for a specific geographical region or a specific product.

Essbase filters are stored in a relational database.

Permission	Description
No Access or None	No inherent access to data values unless access is granted globally
Read	Ability to read data values
Write	Ability to read and update data values
Meta Read	Ability to read metadata (dimension and member names)

You create Essbase filters using Essbase Administration Services (and the MaxL command line), which also enables you to:

- View dimensions and their members.
- List filter operations.
- Validate filters at design time.

See *Oracle Essbase Administration Services Online Help* and *Oracle Essbase Database Administrator's Guide*:

[Oracle Essbase documentation](#)

You can ignore any references to Hyperion Shared Services (HSS) in Enterprise Performance Management documentation.

Configuring Resource Permission Definitions for Essbase Filters

Resource permission definitions combine with Essbase filters in the policy store, and you grant them to an application role. This enables users associated with an application role to secure the data that is defined by one or more combinations of resource permission definitions for Essbase filters.

Before you can grant resource permission definitions for Essbase filters to an application role (see [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#)), the appropriate resource permission definitions must first exist in the policy store. Use the examples in this section to understand how to configure resource permission definitions so that users can use Essbase filters.

An application role requires at least two policy store permission grants to access to a specific filter. You must give an application role permission to use filters within a specific scope.

See [Resource Permissions Reference for Essbase and Associated Tools](#).

Use the following examples to understand how to configure resource permission definitions for Essbase filters:

Example 1 - Configuring resource permissions to enable the use of filters within the Demo application:

This example configures resource permission definitions to enable the use of filters within the Demo application. In this example you must ensure that one of the following resource permission definitions exists in the policy store. For example:

- oracle.essbase.application, /EssbaseCluster-1, use_filter

where:

- oracle.essbase.application, - is the resource type (in this case, an application)
- /EssbaseCluster-1, - is the cluster name
- use_filter - is the action permission (in this case it enables the use of filters)

In this example, the use_filter action permission configures the oracle.essbase.application resource type such that a user associated with this definition in the policy store can use filters in any application in EssbaseCluster-1 (including the Demo application).

OR

- oracle.essbase.application, /EssbaseCluster-1/Demo, use_filter

In this example, the use_filter action permission configures the oracle.essbase.application resource type such that a user associated with this definition in the policy store can use filters in the Demo application in the EssbaseCluster-1.

Example 2 - Configuring resource permissions for specific filters:

This example configures resource permission definitions that enable the use of a specific filter. You must specify additional resource permission definitions that name the filter in scope of the first grant.

For example, to restrict a user's dimensional access in a database called Basic to members defined by the filter called read_filter, the following resource permission definitions are required:

- oracle.essbase.application, /EssbaseCluster-1, use_filter

OR

- oracle.essbase.application, ./EssbaseCluster-1/Demo, use_filter

AND

- oracle.essbase.filter, /EssbaseCluster-1/Demo/Basic/read_filter, apply

In this example, the read_filter action permission configures the oracle.essbase.application resource type such that a user that is associated with this definition in the policy store is restricted to read filters in the Basic database in the Demo application in the EssbaseCluster-1.

Resource Name	Resource Type	Permission Actions	Permission Class
/EssbaseCluster-1	oracle.essbase.server	access	oracle.security.jps.ResourcePermission
/EssbaseCluster-1/Demo	oracle.essbase.application	use_filter	oracle.security.jps.ResourcePermission
/EssbaseCluster-1/Demo/Basic/read_filter	oracle.essbase.filter	apply	oracle.security.jps.ResourcePermission

Example 3 - Configuring resource permissions for multiple filters:

This example activates multiple filters with multiple resource permission definitions to restrict a user's dimensional access in database Basic to members that are defined either by filters "read_filter" or "readFeb_filter." The following resource permission definitions are required:



Note:

This differs from EPM installations where users and groups are limited to a single active filter.

- oracle.essbase.application, /EssbaseCluster-1, use_filter
OR
- oracle.essbase.application, ./EssbaseCluster-1/Demo, use_filter
AND
- oracle.essbase.filter, /EssbaseCluster-1/Demo/Basic/read_filter, apply
AND
- oracle.essbase.filter, /EssbaseCluster-1/Demo/Basic/readFeb_filter apply

Resource Name	Resource Type	Permission Actions	Permission Class
/EssbaseCluster-1	oracle.essbase.server	access	oracle.security.jps.ResourcePermission
/EssbaseCluster-1/Demo	oracle.essbase.application	use_filter	oracle.security.jps.ResourcePermission
/EssbaseCluster-1/Demo/Basic/read_filter	oracle.essbase.filter	apply	oracle.security.jps.ResourcePermission
/EssbaseCluster-1/Demo/Basic/readFeb_filter	oracle.essbase.filter	apply	oracle.security.jps.ResourcePermission

Example 4 - Configuring resource permissions for filters to extend or restrict data access at database level:

This example uses an active filter to extend or restrict data access at the database level.

For example, a user that is associated with the following resource permission definitions cannot read from Demo where noAccess1 restricts access to all dimensions:

- oracle.essbase.application, /EssbaseCluster-1/Demo, read
- oracle.essbase.filter,/EssbaseCluster-1/Demo/Basic/noAccess1, apply

Securing Data Access with Essbase Filters

You secure data access by granting Essbase filters to an application role. An Essbase filter resource permission definition secures data access for the grantee at the database level.

1. If a specific filter does not exist, then create it using Essbase Administration Services Console or the MaxL command line.

See *Oracle Essbase Administration Services Online Help* and *Oracle Essbase Database Administrator's Guide*:

[Oracle Essbase documentation](#)

2. Grant filter resource permission definitions to an application role using Oracle Fusion Middleware.

See [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#).

You can also do this programmatically using Oracle WebLogic Scripting Tool. See Managing Application Policies with OPSS Scripts in *Securing Applications with Oracle Platform Security Services*.

Note:

The following Enterprise Performance Management restriction does not apply when Essbase is installed with Oracle Business Intelligence:

For EPM-installed systems, there can be only one filter per multidimensional database per user or application role. If a user or application role is directly provisioned with a second filter, then the first is revoked. Multiple filters can be provisioned indirectly when a user is a member of multiple application roles that each have a provisioned filter.

Filter resource permission definitions are determined when you connect to a specific Essbase multidimensional database. Filter resource permission definitions pass to the Essbase Agent during authentication. If the user is authenticated successfully, then the list of filters for that user is updated in a locally stored .SEC file.

When Are Filter Access Permission Grant Changes Consumed?

The filter access permissions for a user are determined at login time and are consumed when the Essbase database is selected (SetActive).

Authorization changes are not noticed by existing sessions; a user must log in again to consume changes in authorization policy.

How Do Filter Permission Grants Differ Between Oracle Business Intelligence and EPM?

You grant filter resource permissions differently when Essbase is installed with Oracle Business Intelligence compared to when Essbase is installed as part of an EPM System.

Bear these guidelines while granting filter resource permissions:

- When Essbase is installed with Oracle Business Intelligence.
You grant filter resource permission definitions to an application role, or assign the admin permission to the application role.
- When Essbase is installed as part of an EPM System.
You grant filter resource permissions to a group, or assign the admin permission to the group.

Essbase installed with Oracle Business Intelligence grants a user the union of all permissions that are granted directly or through groups and application roles. Unlike an EPM system, with Oracle Business Intelligence there is no restriction that, for example, the oracle.essbase.application filter action must be granted using the same group as the filters. In Oracle Business Intelligence, you can have conflicting roles. See the following table to understand the consequences of these grants.

Application Role (Group)	Application Grant	Filter Grant
A	oracle.essbase.application, / EssbaseCluster-1/Demo, use_filter	oracle.essbase.filter, /EssbaseCluster-1/Demo/Basic/read_filter, apply
B	oracle.essbase.application, / EssbaseCluster-1/Demo, use_filter	oracle.essbase.filter, /EssbaseCluster-1/Demo/Basic/readFeb_filter, apply

In this table, user JDoe is a member of groups A and B, and so when querying Basic, JDoe has access to rows defined by read_filter and readFeb_filter. If group A has the application grant removed in an EPM System installation, then users of group A no longer have access to any filters. However, when Essbase is installed with Oracle Business Intelligence, user JDoe continues to have access to rows from both filters because JDoe also inherits the permissions from membership of group B.

Configuring Access to Essbase Calculations

Essbase calculations enable you to apply mathematical formula to data in Essbase multidimensional databases. You enable users to access Essbase calculations by configuring resource permission definitions for Essbase calculations and granting them to application roles.

See [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#).

Topics

- [What Are Essbase Calculations?](#)
- [Configuring Resource Permission Definitions for Essbase Calculations](#)

- [Enabling Users to Access Essbase Calculations](#)

What Are Essbase Calculations?

An Essbase calculation enables a user to define and apply complex formulas to dimensional members. You can name calculations and save them in files at the application or database level; these are called calculation scripts. Calculations can also be interactively created and run; these are called inline calculations. Finally, every database has a default calculation defined in the outline.

Calculation scripts are stored in the local file system, and their definitions can be complex and require tool support. For example, you can use Essbase Administration Services (EAS) and Calculation Manager, which provide the ability to:

- View dimensions and their members.
- MaxL can also be used for calculation definition.

MaxL can also be used for calculation definition.

Configuring Resource Permission Definitions for Essbase Calculations

Before you can grant an application role permission to use Essbase calculations appropriate resource permission definitions must exist in the policy store.

See [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#).

Use the examples in this section to understand how to configure resource permission definitions so that users can use Essbase calculations.

See [Resource Permissions Reference for Essbase and Associated Tools](#).

Example 1 - To configure resource permission definitions to use default and inline calculations in /cluster/App1:

This example configures resource permission definitions to use default and inline calculations in /EssbaseCluster-1/App1. The following resource permission definition must exist in the policy store. For example:

- oracle.essbase.application, /EssbaseCluster-1, use_calculation

In this example an application resource permission grants the use_calculation permission to all applications in the cluster.

OR

- oracle.essbase.application, /EssbaseCluster-1/App1, use_calculation

In this example an application resource permission grants the use_calculation permission to applications in App1.

Example 2 - To configure resource permission definitions to use all calculations in /cluster/App1:

This example configures resource permission definitions to use all calculation scripts in /EssbaseCluster-1/App1, you must ensure that the following permissions exist in the policy store. For example:

- oracle.essbase.application, /EssbaseCluster-1, use_calculation

OR

oracle.essbase.application, /EssbaseCluster-1/App1, use_calculation

AND

- oracle.essbase.calculation, /EssbaseCluster-1/App1, all

This calculation permission grants access permissions to use to all calculation scripts in App1.

Example 3 - Configuring resource permission definitions to use all calculations in the cluster:

This example configures resource permission definitions to use all calculations in the cluster, you must ensure that both of the following permissions exist in the policy store. For example:

- oracle.essbase.application, /EssbaseCluster-1, use_calculation
- oracle.essbase.calculation, /EssbaseCluster-1, all

Example 4 - Configuring resource permission definitions to use calculation scripts forecastQ1 and forecastQ2:

This example configures resource permission definitions to use specific calculation scripts in the cluster (for example, forecastQ1 and forecastQ2), you must ensure that the following permissions exist in the policy store. For example:

- oracle.essbase.application, /EssbaseCluster-1, use_calculation
- OR
- oracle.essbase.application, /EssbaseCluster-1/App1, use_calculation
- oracle.essbase.calculation, /EssbaseCluster-1/App1/Db1/forecastQ1, execute
- AND
- oracle.essbase.calculation, /EssbaseCluster-1/App1/Db1/forecastQ2, execute

Note:

A grant to a specific calculation script revokes cluster or application level access to all calculations. Consider specific grants to calculations scripts as restrictions.

For example:

A user with the following grants has access only to the forecastQ1 calculation script:

- oracle.essbase.application, /EssbaseCluster-1, use_calculation
- oracle.essbase.calculation, /EssbaseCluster-1, all
- oracle.essbase.calculation, /EssbaseCluster-1/App1/Db1/forecastQ1, execute

 **Note:**

The presence of an `oracle.essbase.calculation` grant does not imply `oracle.essbase.application calculate` access. For example:

The user does not have access to any calculation, outline, inline, or script with any of following grants if there is no `oracle.essbase.application calculate` grant:

- `oracle.essbase.calculation, /EssbaseCluster-1/App1, all`
- `oracle.essbase.calculation, /EssbaseCluster-1, all`
- `oracle.essbase.calculation, /EssbaseCluster-1/App1/Db1/forcastQ1, execute`

Enabling Users to Access Essbase Calculations

You enable a user to access Essbase calculations by granting one or more Essbase calculation access permissions to an application role that is associated with the user.

By default, users with the `calculate` permission can execute the default and inline calculations on all databases.

1. If the calculation script does not exist, then create and activate it using Essbase Administration Services Console or the MaxL command line.

See *Oracle Essbase Database Administrator's Guide and Oracle Essbase Administration Services Online Help*:

[Oracle Essbase documentation](#)

2. Grant the Essbase calculation access permission to an application role using Fusion Middleware Control.

See [Enabling Users to Perform Specific Actions in Essbase and Associated Tools](#).

Changing Essbase Ports in Oracle Business Intelligence

Essbase Agent runs on two types of ports. Since Agent runs on BI Managed WebLogic Server, it has an HTTP Port that it listens to.

Since this port is common to all the applications hosted on the BI Managed server, this section covers the changing of Essbase specific ports.

By default, Essbase Agent uses port 9799. To change this port, edit the `AGENTPORT` parameter in `essbase.cfg`.

The Essbase Server uses a port range and by default it uses the available ports. To explicitly specify a server port range, define `SERVERPORTBEGIN` and `SERVERPORTEND` settings in `essbase.cfg`.

Use these steps to update the Essbase Agent port number or Essbase server port range.

1. To update the Essbase Agent port number.
 - a. Create a backup of `DOMAIN_HOME/config/fmwconfig/biconfig/essbase/essbase.cfg`

- b. Open `essbase.cfg`, and locate and update the `AGENTPORT` parameter.
For example: `AGENTPORT 9799` can be changed to `AGENTPORT 9779` where `9779` is any free port of your choice.
 - c. Save the `essbase.cfg` file.
 - d. Restart the BI Managed server.
2. To update the Essbase server port range.
 - a. Open the `essbase.cfg` file for editing:
`DOMAIN_HOME/config/fmwconfig/biconfig/essbase/essbase.cfg`
 - b. Add `SERVERPORTBEGIN` and `SERVERPORTEND` parameters to suit your environment.
For example:
`SERVERPORTBEGIN 9000`
`SERVERPORTEND 9499`
 - c. Save the `essbase.cfg` file.
 - d. Restart the BI Managed server.

 **Note:**

If you are performing this step after a scaleout, make sure to edit the corresponding `essbase.cfg` on all other nodes, to have the same entries as primary and restart the BI Cluster (all the managed servers of `bi_cluster`).

Resource Permissions Reference for Essbase

This section provides reference information about the resource permissions for Essbase that are configured by default when you install Essbase with Oracle Business Intelligence.

To access Essbase-related functionality, a user (or group, or application role that the user belongs to), needs to be granted one or more resource permissions. Essbase-related resource permissions are defined by resource type, name, and actions and are held in the Policy Store as part of the Oracle Fusion Middleware security model. See *Managing the Policy Store in Securing Applications with Oracle Platform Security Services*.

- **Resource Type**
Includes a named group of permissions.
- **Resource Name**
Specifies the scope for which a permission applies.
- **Action**
Defines what operation the permission allows the grantee to perform.

You manage Essbase-related resource permissions using Oracle Fusion Middleware Control. However, you can also use APIs in Oracle Platform Security Services using JMX or Oracle WebLogic Scripting Tool.

 **Note:**

When Essbase is installed with Oracle Business Intelligence, the functionality described in this chapter replaces that provided by Hyperion Shared Services for provisioning users.

This topic contains the following sections:

- [What Resource Types Apply to Essbase and Associated Tools?](#)
- [What Resource Names Apply to Essbase and Associated Tools?](#)
- [What Actions Apply to Essbase and Associated Tools?](#)

What Resource Types Apply to Essbase?

Resource types are named groups of permissions that can be associated with specific actions. There are several levels of authority in the Essbase server with a resource type for each. Resource types are used by Fusion Middleware Control to limit the list of applicable Essbase-related actions when selecting a new grant.

The following table lists the Essbase-related resource types that are supported in Oracle Business Intelligence.

Resource Type	Description
oracle.essbase.server	Global, cluster, and server level permissions.
oracle.essbase.application	Application level permissions required to use filters and calculations.
oracle.essbase.filter	Filter access control.
oracle.essbase.calculation	Calculation script access control.

What Resource Names Apply to Essbase?

Each resource type has a set of resources and actions that can be authorized. Essbase-related resource names are either specific objects or scopes in which objects are contained.

For oracle.essbase.server and oracle.essbase.application resource types, resource names are scopes. They are hierarchical and as such any scope in the following table includes the set of scopes contained within it.

The following table lists the supported Essbase scopes:

Name	Scope
/	Global, all clusters, all applications, all cubes.
/cluster	All applications within a logical cluster.
/cluster/application	Specific application and its cubes.

The following table lists the supported oracle.essbase.filter scopes:

Name	Scope
/cluster/application/ database/filtername	Access to a specific, named filter.

The following table lists the supported Essbase calculation scopes:

Name	Scope
/cluster/	All calculation scripts at cluster level: any applications, any cubes.
/cluster/application	All calculation scripts at application level: any cube within the named application.
/cluster/application/ database/scriptname	Access to a specific named calculation script.

What Actions Apply to Essbase?

Essbase-related actions (permissions) represent operations that a user can perform on Essbase.

Actions granted to `oracle.essbase.server` or `oracle.essbase.application` resource types are hierarchical, such that any action listed includes the set of permissions listed beneath it.

For example, granting `oracle.essbase.application, /EssbaseCluster-1, read` enables the grantee to read multidimensional databases and to restart the applications in `EssbaseCluster-1`.

Actions on other Essbase-related resource types are not hierarchical and need to be individually granted.

The series of tables below give details on the actions available with each Essbase-related resource type.

The following table lists the supported `oracle.essbase.server` actions:

Action	Description
administer	Full access to administer the server, applications, and databases.
Create	Ability to create and delete applications and databases within applications. Includes Application Manager and Database Manager permissions for the applications and databases created by this user.
Access	Ability to log in to Essbase. This is deprecated and the existence of any server or application permission now suffices.

Note:

A user that creates an application has permission to manage that application within the same session. The same user requires an explicit grant to manage the application in subsequent sessions.

The following table lists supported oracle.essbase.application actions.

Action	Description
manage_application	Ability to create, delete, and modify databases and application settings within the particular application. Includes Database Manager permissions for databases within the application.
manage_database	Ability to manage databases (for example, to change the database properties or cache settings), database artifacts, locks, and sessions within the assigned application.
use_calculation	Ability to execute calculations, update, and read data values based on the assigned scope, using any assigned calculations and filter.
write	Ability to update and read data values based on the assigned scope, using any assigned filter.
read	Ability to read data values.
use_filter	Ability to access specific data and metadata according to the restrictions of a filter.
restart	Ability to start and stop an application or database.

The following table lists supported oracle.essbase.filter actions.

Action	Description
Apply	Apply the filter identified by the resource name.

The following table lists supported oracle.essbase.calculation actions.

Action	Description
all	Enables the user to execute any calculation that is contained in the scope referenced by the resource name.
execute	Enables the user to execute the calculation script that is identified by the resource name.

The following table lists permissions granted to Oracle Business Intelligence application roles.

Role Name	Resource Permission	Role Members
BIAdministrator	oracle.essbase.server, /, administrator	BIAdministrators group
BIAuthor	not applicable	BIAuthors group BIAdministrator application role
BIConsumer	oracle.essbase.server,/,access,use_filter	BIConsumers group BIAuthor application role
AuthenticatedUser	not applicable	BIConsumer

This application role hierarchy and permission grants are defaults only and the administrator user can change them in Fusion Middleware Control.

AuthenticatedUser is a member of BIConsumer by default. This means that any successfully authenticated user has BIConsumer role permissions.

Managing Essbase System Administration in Oracle Business Intelligence

You manage Essbase system administration in Oracle Business Intelligence by starting and stopping Essbase Agents, enabling and viewing log files, setting logging levels, migrating Essbase configuration between domains, monitoring Essbase metrics, and backing up and recovering Essbase data.

This topic contains the following sections:

- [Starting and Stopping Essbase Components](#)
- [Maintaining High Availability of Essbase Components in Oracle Business Intelligence](#)
- [Configuring Logging for Essbase Components](#)
- [Migrating Essbase Configuration Between Domains](#)
- [Monitoring Essbase Metrics](#)
- [Backup and Recovery of Essbase Data](#)

Starting and Stopping Essbase Components

You can monitor status, and start and stop Essbase components using BI process control commands (uses JAgent).

See [Managing Oracle Business Intelligence Processes](#) .

Maintaining High Availability of Essbase Components in Oracle Business Intelligence

High availability for Essbase in Oracle Business Intelligence is automatically maintained using an active-passive fault tolerance model.

Scaling Out Essbase to Support High Availability

You scale out Essbase onto additional computers to support high availability between computers. Scale-out is achieved in a similar way to existing Oracle BI EE components (see [Managing Availability in Oracle Business Intelligence \(Horizontally Scaling\)](#)).

About the Essbase Active-Passive Topology

The two-node active-passive topology applies to the Essbase Server and Agent plus the mid-tier Essbase Administration Services, Analytic Provider Services, and Essbase Studio Server components.

See Oracle Essbase High Availability Concepts, Configuring Oracle Essbase Clustering, and Oracle Hyperion Provider Services Component Architecture in *High Availability Guide*.

Managing Essbase Capacity

You manage Essbase capacity within Oracle Business Intelligence by monitoring Essbase components and service levels using Fusion Middleware Control to answer the following questions:

- How do I know that Essbase components are running?
See [Displaying Oracle Business Intelligence Pages in Fusion Middleware Control](#).
- How do I know if my system is overloaded?
See [Monitoring Service Levels](#).

Configuring Logging for Essbase Components

You configure logging for all Essbase components in the BI Domain logging.xml file.

The file is located in:

```
DOMAIN_HOME/config/fmwconfig/servers/<BI_MANAGED_SERVER_NAME>
```

Log configuration details for each Essbase subcomponent, along with the location of the corresponding Oracle Diagnostic Log (ODL) files are provided below:

- [Essbase Java-Agent specific Logging Configuration](#)
- [Essbase Applications specific Logging Configuration](#)
- [Provider Services \(APS\) specific Logging Configuration](#)
- [Essbase WebServices specific Logging Configuration](#)
- [Essbase Cube Deployment Service specific Logging Configuration](#)

Note:

To capture every possible level of logging information for a particular component, set the logger level field value to 'TRACE:32'.

Essbase Java-Agent specific Logging Configuration

```
<log_handler name='jagent-handler-text'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
  <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/essbase/
jagent.log' />
  <property name='maxFileSize' value='10485760' />
  <property name='maxLogSize' value='104857600' />
  <property name='useSourceClassAndMethod' value='true' />
</log_handler>
<logger name='oracle.JAGENT' level='NOTIFICATION:1' useParentHandlers='false'>
  <handler name='jagent-handler-text' />
</logger>
```

Essbase Applications specific Logging Configuration

```
<log_handler name='serverhandler' class='oracle.core.ojdl.logging.ODLHandlerFactory'>
  <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/
essbase/essbase' />
```

```

        <property name='maxFileSize' value='10485760' />
        <property name='maxLogSize' value='524288000' />
    </log_handler>
    <logger name='DefSvrLogger' level='TRACE:1' useParentHandlers='false'>
        <handler name='serverhandler' />
    </logger>

```

Provider Services (APS) specific Logging Configuration

```

<log_handler name='provider-services-handler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
    <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/aps/
apserver.log' />
    <property name='maxFileSize' value='10485760' />
    <property name='maxLogSize' value='104857600' />
</log_handler>
<logger name='oracle.EPMOHPS' level='WARNING' useParentHandlers='false'>
    <handler name='provider-services-handler' />
</logger>

```

Essbase WebServices specific Logging Configuration

```

<log_handler name='essbase-ws-handler'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
    <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/
essbasews.log' />
    <property name='maxFileSize' value='10485760' />
    <property name='maxLogSize' value='104857600' />
</log_handler>
<logger name='oracle.EPMOHEWS' level='WARNING:1' useParentHandlers='false'>
    <handler name='essbase-ws-handler' />
</logger>

```

Essbase Cube Deployment Service specific Logging Configuration

```

<log_handler name='cds-handler' class='oracle.core.ojdl.logging.ODLHandlerFactory'>
    <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/cds/
cds.log' />
    <property name='maxFileSize' value='10485760' />
    <property name='maxLogSize' value='104857600' />
</log_handler>
<logger name='oracle.essbase.cds' level='NOTIFICATION:1' useParentHandlers='false'>
    <handler name='cds-handler' />
</logger>

```

Migrating Essbase Configuration Between Domains

You can migrate an Essbase configuration between domains for an Oracle Enterprise Performance Management System installation.

See *Moving Oracle Hyperion Enterprise Performance Management System to a Target Environment* in *Administering Oracle Fusion Middleware*.

Monitoring Essbase Metrics

You monitor Essbase metrics using Fusion Middleware Control.

See [Monitoring Service Levels](#).

Backup and Recovery of Essbase Data

This section describes backing up and recovering Essbase data when Essbase is installed with Oracle Business Intelligence.

For information about backing up and recovering Essbase data, see:

- Oracle Business Intelligence and Essbase - When installed using the Oracle Business Intelligence installer, see Backup and Recovery Recommendations for Oracle Essbase in *Administering Oracle Fusion Middleware*.
- EPM System products - When installed using the EPM System Installer, see *Oracle Enterprise Performance Management System Backup and Recovery Guide* at:

<http://www.oracle.com/technetwork/middleware/performance-management/documentation/index.html>

Working with Essbase Data in Oracle Business Intelligence

These topics provide orientation for using Essbase in Oracle Business Intelligence.

This section introduces working with Essbase cubes in Oracle Business Intelligence:

- [Enabling Single Sign-On for Essbase Data Sources](#)
- [Creating, Scheduling, and Running Analyses and Reports Where Essbase Is the Data Source](#)
- [Enabling Oracle BI EE to Connect to Essbase Over SSL](#)

Enabling Single Sign-On for Essbase Data Sources

To enable SSO for Essbase data sources installed using the Oracle Business Intelligence installer, you select **SSO** in the General tab of the connection pool object that corresponds to the Essbase data source in the Oracle BI repository.

Also select the **Virtual Private Database** option in the corresponding database object to protect cache entries.

See Multidimensional Connection Pool Properties in the General Tab in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Also see note 1993210.1 at:

<https://support.oracle.com>

Creating, Scheduling, and Running Analyses and Reports Where Essbase Is the Data Source

A user with the BI Author application role can create, schedule, and run analyses and reports where Essbase is the data source, where filters are applied using the identity of the end user, and where SSO is configured.

For more information about:

- Creating, scheduling, and running Oracle Business Intelligence analyses, see:
User's Guide for Oracle Business Intelligence Enterprise Edition
- Creating and running BI Publisher reports, see:
Administrator's Guide for Oracle Business Intelligence Publisher

Enabling Oracle BI EE to Connect to Essbase Over SSL

This section addresses how Oracle BI Server handles Essbase connectivity when both the Oracle BI Server and Essbase are installed and configured as part of the Oracle Business Intelligence.

If Essbase is configured for SSL, the BI Server (which uses the Essbase RTC client), must perform all the client side SSL configuration as an Essbase client.

If the BI Server and Essbase run on the same machine, and the Essbase server is configured for SSL, then the BI Server can simply point to the Essbase configuration file (`essbase.cfg`). To do this you add the `ESSBASE_CONFIG_PATH` property to `obis.properties` to specify the location of `essbase.cfg`.

If you need the Oracle BI Server to have a custom client side `essbase.cfg` (for example, to control specific Essbase client settings), but you do not want to use the Essbase server configuration file, then refer to the sections "Setting up an Essbase Client Wallet" and "Copying and Configuring the Wallet Path" in the 12c version of *Essbase Database Administrator Guide*. In this case, the `obis.properties` file must point to the directory location which contains `essbase.cfg` defined for the Oracle BI Server.

To make the BI Server to point to the `essbase.cfg` location:

1. Open the `obis.properties` file located here:

```
DOMAIN_HOME/config/fmwconfig/bienv/OBIS/obis.properties
```

Add the `ESSBASE_CONFIG_PATH` property with a value that either represents the path to location of `essbase.cfg` defined for Oracle BI EE, or the location that is already being used by Essbase Server.

For example, to point to the Essbase server's configuration file, then:

```
ESSBASE_CONFIG_PATH=DOMAIN_HOME/config/fmwconfig/biconfig/essbase
```

2. Save the `obis.properties` file.

Note:

End-to-End SSL configuration in 12.2.1.0.0 release is not supported. That is, if the Oracle BI Server is also configured for SSL, then connectivity between Oracle BI Server and the Essbase server will not work. However, this will be supported in a future release of 12c.

Where Can I Learn More Information About Essbase?

There are several places where you can learn more about Essbase.

Use the following links to learn more about Essbase in Oracle Enterprise Performance Management System.

- For all Essbase and EPM documentation, use the Performance Management Documentation page:

<http://www.oracle.com/technetwork/middleware/performance-management/documentation/index.html>

This page also contains links to EPM System Supported Platform Matrices, My Oracle Support, and other information resources.

Drill down to view specific documents for a release.

- For information about meeting system requirements and understanding release compatibility, use *Oracle Enterprise Performance Management System Certification Matrix*:

<http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html>

- When Essbase is installed as part of Oracle BI Enterprise Edition, refer to *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition* for information about system requirements and release compatibility.

Part IX

Reference Information

There are several resources with reference information for managing Presentation Services.

It includes the following appendixes:

- [Configuration File Settings](#)
- [Advanced Configuration Reference](#)
- [Mapping User Interface Labels with Configuration File Elements](#)
- [BI-Specific WLST Command Reference](#)

A

Configuration File Settings

This section lists key Oracle Business Intelligence system configuration files and provides details about NQSConfig.INI file parameters. This appendix includes the following sections:

- [Configuration Files](#)
- [NQSConfig.INI File Configuration Settings](#)

Configuration Files

Oracle Business Intelligence configuration files control the behavior of the system.

The table below lists key Oracle Business Intelligence configuration files 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: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig/OBIS
Oracle BI Server	instanceconfig.xml credentialstore.xml marketingwebexpressions.xml userpref_currencies.xml	For example: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig/OBIPS Do not add elements to the instanceconfig.xml file unless you are 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: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig/OBICCS
Oracle BI Scheduler	instanceconfig.xml schedulerconfig.xml	For example: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig/OBISCH
JavaHost	config.xml logging_config.xml	For example: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig/OBIJH
Oracle BI Presentation Services Plug-in	bridgeconfig.properties	For example: <i>BI_DOMAIN</i> /config/fmwconfig/ biconfig

BI Component	Configuration File	File Location
Oracle BI Environment	bi-environment.xml odbc.ini obis.properties	<p>For example:</p> <p><i>BI_DOMAIN</i>/config/fmwconfig/bienv/core</p> <p>bi-environment.xml contains BI-specific environment configuration settings (separate from process command-specific parameters), for example:</p> <ul style="list-style-type: none"> Data directory (SDD) - path to a singleton directory (for high availability purposes). By default, SDD = \$DOMAIN_HOME/bidata. As with 11g, it is 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 do not), is not allowed. See Key Directories in Oracle Business Intelligence. 11g 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 do not display hostname), it may be required to change these during cloning operations. <p>odbc.ini contains the single source of truth for ODBC connection endpoints:</p> <ul style="list-style-type: none"> 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. <p>obis.properties contains BI Server-specific environment and configuration settings that are substituted with specified values, and is located in:</p> <p><i>BI_DOMAIN</i>/config/fmwconfig/bienv/OBIS</p>

See *Installing and Configuring Oracle Business Intelligence*.

NQSCONFIG.INI File Configuration Settings

This section lists the NQSCONFIG.INI file parameters for Oracle Business Intelligence and gives a brief description and any required syntax for each parameter. The Oracle BI 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 are editing a Windows version of NQSSConfig.INI. If you are editing this file on a UNIX system, then ensure that you use UNIX-appropriate file system paths and conventions.

This topic includes the following sections:

- [About Parameters in the NQSSConfig.INI File](#)
- [Repository Section Parameters](#)
- [Multitenancy Section Parameters](#)
- [Query Result Cache Section Parameters](#)
- [General Section Parameters](#)
- [Security Section Parameters](#)
- [Server Section Parameters](#)
- [High Availability Parameters](#)
- [Dynamic Library Section Parameters](#)
- [Usage Tracking Section Parameters](#)
- [Query Optimization Flags Section Parameters](#)
- [MDX Member Name Cache Section Parameters](#)
- [Aggregate Persistence Section Parameters](#)
- [JavaHost Section Parameters](#)
- [Datamart Automation Section Parameters](#)

About Parameters in the NQSSConfig.INI File

The Oracle BI Server has one NQSSConfig.INI file.

Note the following rules and guidelines for NQSSConfig.INI file entries:

- The Oracle BI Server reads the NQSSConfig.INI file each time it is started.
- Each parameter entry in NQSSConfig.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 NQSSConfig.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 the Oracle BI 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 the Oracle BI Server from starting. The errors are logged to the `nqserver.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 the Oracle BI 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 Oracle Business Intelligence Processes](#).

Repository Section Parameters

The Repository section contains one entry for every repository that is loaded when the server starts.

Note:

Hosting multiple repositories on a single Oracle BI Server is not 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 repository. Client tools use this name to configure the ODBC data sources that connect to the repository. 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 repository. The file name must have the .rpd file extension, and the file must reside in the repository subdirectory.

The demonstration repository SampleAppLite.rpd is installed when selected at install time with Oracle Business Intelligence.

When `DEFAULT` is specified for a repository, connections that do not specify a logical repository name in the DSN connect to the default repository.

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 executed 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 Business Intelligence, including information about how to use agents to seed the Oracle BI Server cache, see [Managing 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 [Using 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 does not start with "/" (on UNIX) 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 do not 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 are not 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 have 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 is not cached.

When set to 0, there is 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 [Using 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 [Using 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 executed query, then the new query is not 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 executed query (in this example, by aggregating data from the first result set stored in the cache). That is, Oracle Business Intelligence 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 is not 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 are not affected by this parameter. If a query result is satisfied by the cache—that is, the query gets a cache hit—then this query is not 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 did not 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 is not 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 did not 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 did not 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 is not added to the cache.
- **Noncacheable table.** Physical tables in the Oracle BI Server repository can be marked "noncacheable." If a query references any noncacheable table, then the query results are not 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 are not 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 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 [Configuring 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 [Using 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 Business Intelligence, 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 Oracle BI Catalog Manager and language extensions, see [Localizing Oracle Business Intelligence](#) .

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 is defined in the Physical layer in the Oracle BI 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 Oracle BI 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

Functional Category	Specific Parameters
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 does not match the SORT_ORDER_LOCALE setting in the Features tab of the Database dialog in the Oracle BI repository, then result sets might not be correct. If the settings do not 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 does not match the SORT_ORDER_LOCALE setting in the Features tab of the Database dialog in the Oracle BI repository, then query performance might be negatively impacted. However, this situation does not affect the correctness of the result set.

```
SORT_ORDER_LOCALE = "english-usa";
```

SORT_ORDER_LOCALE on UNIX 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 [Setting 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 [Making 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 is not 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 are not 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 has not 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` does not 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 `NQSSConfig.INI` file, the parameter applies to all back-end databases in a repository. Therefore, set the parameter to match the case sensitivity of the dominant back-end database of the repository.

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` is not 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 do not affect the format of the timestamps written to the `nqserver.log` or the `nquery.log`. The format of the timestamps written in the log files is fixed according to Oracle Fusion Middleware standards and cannot 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 UNIX) 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.

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 NT 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 NT system buffers, as each I/O still goes through the system buffers. 128 KB is also a reasonable value on UNIX 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 Oracle BI repository object privilege granted to the `AuthenticatedUser` application role, which is the default application role associated with any new repository object.

In effect, this setting specifies the default level of object security in the Presentation layer of the repository for new objects that do not have other explicit security settings.

Note that the `AuthenticatedUser` application role means "any authenticated user." This role is internal to the Oracle BI repository.

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 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 is not 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 cannot 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 execute 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 execution 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 execute. If after executing the fifth initialization block the ten minute login maximum is exceeded, then the Oracle BI Server does not execute 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 will log 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 Business Intelligence components.

The default setting for `SSL` is `NO`.

See SSL Configuration in Oracle Business Intelligence in *Security Guide for Oracle Business Intelligence Enterprise Edition* for information about configuring SSL between Oracle Business Intelligence components.

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 Business Intelligence 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. See *Upgrade Guide for Oracle Business Intelligence*

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 Oracle BI repository files when the Administration Tool is in either online or offline mode.

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 [Using Fusion Middleware Control to Disallow RPD Updates](#).

The default is `NO`, meaning that repositories can be edited.

When this parameter is set to `YES`, it prevents the Administration Tool from making any changes to repository files. When the Administration Tool opens the repository, a message informs the user that the repository is read-only. If this parameter is set to `NO`, then the Administration Tool can make changes to the repository.

Even when `READ_ONLY_MODE` is set to `NO`, there are still situations when Administration Tool opens repositories in read-only mode. For example, if you open a repository in offline mode, but the Oracle BI Server or another Administration Tool client holds a lock on the repository, then the repository opens in read-only mode. In online mode, a repository might open as read-only if an offline Oracle BI Server held a lock on the repository at the time the Oracle BI Server started.

In addition, the Administration Tool also opens in read-only mode when Oracle Business Intelligence has been clustered, and the Administration Tool is connected in online mode to a slave node. This occurs because the master node holds a lock on the repository. To avoid this situation when running in a clustered environment, ensure that the Oracle BI Server ODBC DSN that is used by the 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 500 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 = 500;`

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 Business Intelligence 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 Business Intelligence 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 execute 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. Do not 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 repository. 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. Do not raise this value without consulting Oracle Support Services.

Example: `MAX_DRILLDOWN_INFO_CACHE_ENTRIES = 1024;`

MAX_DRILLDOWN_QUERY_CACHE_ENTRIES

Controls the number of cached Action Link query entries per repository. 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. Do not raise this value without consulting Oracle Support Services.

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 Business Intelligence, it is recommended that you do not 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, do not 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 reference 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 does not execute a query. If you specify 1 in this parameter, then only queries that do not 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;`

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 Business Intelligence 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 `nqserver.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 Business Intelligence 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 release.

LISTEN_PORT

This parameter is reserved for a future release.

ENABLE_DB_HINTS

Enables optional hints to be passed along with a SQL statement to an Oracle Database. Database hints are discussed in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

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 Business Intelligence 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 [Deploying Oracle Business Intelligence for High Availability](#).

Example: `CLUSTER_PARTICIPANT = YES;`

DISCONNECTED

This parameter has been deprecated and is no longer used.

AUTOMATIC_RESTART

Specifies whether the Oracle BI Server is automatically restarted after a failure.

Automatic restart applies only to an Oracle BI Server platform; it does not apply to a clustered Oracle BI Server environment. The default value is `YES`.

Example: `AUTOMATIC_RESTART = YES;`

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.

See Database Functions in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for more information about the `EVALUATE*` functions.

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 metadata repository.

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. Do not 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 Oracle BI repository from Oracle Database and TimesTen has decimal/numeric data set to NUMERIC, and decimal/numeric SQL code that is 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 executed by the Oracle BI Server, enable the NUMERIC_SUPPORTED database feature in the Physical layer database object. See *Specifying SQL Features Supported by a Data Source in Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

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 Business Intelligence, 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 is not 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 release.

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 UNIX 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:

Do not 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 release:

- ODBC200 = nqsdbgatewayodbc;
- ODBC350 = nqsdbgatewayodbc35;
- OCI8 = nqsdbgatewayoci8;
- OCI8i = nqsdbgatewayoci8i;
- OCI10g = nqsdbgatewayoci10g;
- DB2CLI = nqsdbgatewaydb2cli;
- DB2CLI35 = nqsdbgatewaydb2cli35;
- NQSXML = nqsdbgatewayxml;

- XMLA = nqsdbgatewayxmla;
- BAPI = nqsdbgatewaysapbapi;
- ESSBASE = nqsdbgatewayessbasecap;
- 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 repository.

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
ERROR_TEXT	Varchar(250)	Error flag and reason text for queries that do not 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 executed the query)	Null if the request is not run as an impersonated user.

For more information about usage tracking, see [Managing Usage Tracking](#).

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 [Managing Usage Tracking](#).

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 [Managing Usage Tracking](#).

- 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	STORAGE_DIRECTORY	"full_directory_path"
NO	CHECKPOINT_INTERVAL_MINUTES	5
NO	FILE_ROLLOVER_INTERVAL_MINUTES	30
NO	CODE_PAGE	"ANSI"

DIRECT_INSERT Setting	Parameters Used	Parameter Setting
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.

The parameter `STORAGE_DIRECTORY` is valid only if the parameter `DIRECT_INSERT` is set to `NO`.

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 repositories, 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 repositories 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 Administration Tool.

Note:

For new (non-upgraded) installations, the `PHYSICAL_TABLE_NAME` parameter can be updated by editing `NQSConfig.INI`.

See [Managing Usage Tracking](#).

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 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 [Managing Usage Tracking](#).

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 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 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 will stop and wait 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 Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* .

Enables or disables the collection of Oracle BI 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 are running Oracle Business Intelligence on the Oracle Exalytics Machine. See *Using Oracle BI Summary Advisor to Identify Query Candidates for Aggregation in Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* .

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 Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

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 Administration Tool.

Example: `SUMMARY_ADVISOR_TABLE_NAME = "Orcl"."DEV_BIPLATFORM".
"S_NQ_SUMMARY_ADVISOR"`

Query Optimization Flags Section Parameters

There is one parameter in the Query Optimization Flags section.

It is a special parameter to override the behavior of the Oracle BI Server in certain situations.

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.

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;`

MDX Member Name Cache Section Parameters

The parameters in this section are for a cache subsystem that maps between a unique name and the captions of members of all SAP/BW cubes in the repository.

ENABLE

This parameter indicates if the feature is enabled or not.

The default value is `NO` because this only applies to SAP/BW cubes.

DATA_STORAGE_PATH

The path to the location where the cache is persisted. This applies only to a single location.

The number at the end of the entry indicates the storage capacity. When the feature is enabled, the string `<full directory path>` must be replaced with a valid path.

Example: `DATA_STORAGE_PATH = "C:\OracleBI\server\Data\Temp\Cache" 500 MB;`

MAX_SIZE_PER_USER

The maximum disk space that is allowed for each user for cache entries.

Example: `MAX_SIZE_PER_USER = 100 MB;`

MAX_MEMBER_PER_LEVEL

The maximum number of members in a level that can be persisted to disk.

Example: `MAX_MEMBER_PER_LEVEL = 1000;`

MAX_CACHE_SIZE

The maximum size for each individual cache entry size.

Example: `MAX_CACHE_SIZE = 100 MB;`

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 executed 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";

JVAHOST_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 Business Intelligence 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 Business Intelligence running on a Windows computer to access Hyperion Financial Management even if the main instance of Oracle Business Intelligence 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:

JVAHOST_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 release.

ESSBASE_SERVER

This parameter is reserved for a future release.

DMA_DATABASE

This parameter is reserved for a future release.

B

Advanced Configuration Reference

This appendix describes advanced postinstallation configuration and administration procedures that are *not* specific to analyses, agents, dashboards, or the Oracle BI Presentation Catalog. Directions for configuring these components of Oracle Business Intelligence are in earlier chapters. Most administrators need not change the configuration settings that are described in this appendix.

This appendix includes the following sections:

- [Making Advanced Configuration Changes for Presentation Services](#)
- [Using the JavaHost Service for Oracle BI Presentation Services](#)

Making Advanced Configuration Changes for Presentation Services

The Oracle BI 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 UNIX, the process is `sawserver`

The `instanceconfig.xml` file stores the configuration settings that affect Oracle BI Server. Many configuration settings are available in Fusion Middleware Control and that is the preferred method for making configuration changes. If a particular setting is not 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 Oracle BI 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 path to the Oracle BI Presentation Catalog, and the name of the Oracle Business Intelligence Server data source used by Presentation Services to access Oracle BI Server.

Note:

If you have 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>
  <Catalog>
    <ServiceInstanceQuota>
      <Enabled>true</Enabled>
      <SizeLimitMB>8192</SizeLimitMB>
    </ServiceInstanceQuota>
  </Catalog>
  <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>
    <EnableSavingContentWithHTML>true</EnableSavingContentWithHTML>
  </Security>
  <ODBC>
    <UnaccessedRunningTimeoutMinutes>5</UnaccessedRunningTimeoutMinutes>
  </ODBC>
  <UI>
    <MaxSearchResultItemsToReturn>300</MaxSearchResultItemsToReturn>
    <UserPickerDialogMaxAccounts>300</UserPickerDialogMaxAccounts>
  </UI>
</ServerInstance>
```

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

Element	Description	Default Value
<code><ClientStorage></code>	Specifies the parent element for maintaining client state across sessions and within a session. See also the <code>Enabled</code> , <code>LocalStorage</code> , and <code>SessionStorage</code> elements.	No default value

Element	Description	Default Value
<pre><ClientStorage> <Enabled> </ClientStorage></pre>	<p>Specifies whether to maintain client state across sessions and within a session. State is not maintained across browser sessions. The following items maintain state:</p> <ul style="list-style-type: none"> 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. <p>See also the <code>LocalStorage</code> and <code>SessionStorage</code> elements.</p>	true
<pre><ClientStorage> <LocalStorage> </ClientStorage></pre>	<p>Specifies whether the local storage of the browser is used to maintain state. If the browser does not support local storage, then no state is maintained.</p>	true
<pre><ClientStorage> <SessionStorage> </ClientStorage></pre>	<p>Specifies whether the local storage of the browser is used to maintain state for sessions. If the browser does not support local storage, then no state is maintained.</p>	true
<pre><FavoritesSyncUpIdleSeconds></pre>	<p>Specifies the number of seconds of idle time before synchronizing data from a mobile application and favorites from the Oracle BI Presentation Catalog.</p>	300
<pre><BIClientInstallerURL64Bit></pre>	<p>Specifies that you want to override the default download location for the Oracle BI Client Installer when the user selects to download the Oracle BI Client Installer from the Oracle BI EE Home page. The file for the 64-bit Installer is named <code>biee_client_install64.exe</code>.</p>	No default value
<pre><Catalog> <ServiceInstanceQuota> </Catalog></pre>	<p>Specifies the catalog size limit.</p> <p>If you encounter an error similar to <code>Service instance quota has been exhausted</code>, you have exceeded the default limit of 5GB. You can use this parameter to either disable the size limit or to increase it.</p> <p>If you are migrating from 11g which had no size limit, you may encounter an error message similar to <code>Service instance hard quota has been exhausted</code> in the presentation server log. Increase the size limit or disable this parameter and then migrate your catalog again as described in support note 2079953.1 on My Oracle Support at https://support.oracle.com.</p>	No default value

Element	Description	Default Value
<pre><Catalog> <ServiceInstanceQuota> <Enabled> </ServiceInstanceQuota> </Catalog></pre>	<p>Specifies that you want to enable a catalog size limit.</p> <p>If you set the parameter to false, then you remove the quota limit.</p>	true
<pre><Catalog> <ServiceInstanceQuota> <SizeLimitMB> </ServiceInstanceQuota> </Catalog></pre>	Specifies the new size limit in MB for the catalog.	5000
<pre><Security> <AllowRememberPassword> </Security></pre>	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
<pre><Security> <CookieDomain> </Security></pre>	Specifies the domain information for a cookie that is sent to the browser.	No default value
<pre><Security> <CookiePath> </Security></pre>	Specifies the domain path to which cookies apply.	/analytics
<pre><Security> <InIFrameRenderingMode> </Security></pre>	See Protecting Pages in Oracle BI EE from Attack for information.	sameDomainOnly
<pre><Security> <LogonExpireMinutes> </Security></pre>	Specifies a time in minutes after which an inactive user is automatically logged off.	180
<pre><Security> <EnableSavingContentWithHTML> </Security></pre>	<p>Setting this to true enables the Contains HTML Markup option to be displayed in several dialogs and editors across Presentation Services.</p> <p>You must also enable the corresponding privileges in the Manage Privileges page as described in <code>EnableSavingContentWithHTML</code>.</p>	false
<pre><Security> <cursors> <NewCursorWaitSeconds> </cursors> </Security></pre>	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 are not returning in 1 second.	No default value

Element	Description	Default Value
<pre><ODBC> <UnaccessedRunningTimeoutMinutes> </ODBC></pre>	<p>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.</p> <p>This element addresses the case where a user is editing an analysis and browses elsewhere, abandoning the analysis, at least temporarily. Do not set the value too small, however, as the user might return to the analysis.</p> <p>Use this element only for Presentation Services queries that run against the BI Server. The element does not apply to any other type of connection.</p>	5
<pre><UI> <MaxSearchResultItemsToReturn> </UI></pre>	<p>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.</p>	300
<pre><UI> <UserPickerDialogMaxAccounts> </UI></pre>	<p>Specifies the maximum number of items to display in the left picker within a directory listing of the catalog within Presentation Services. For example, a catalog folder with more than 300 items might not display them all unless you increase this value to greater than 300. The minimum value is 0. Use care when setting this element to a high value as the performance of the user interface might decrease.</p>	300

Protecting Pages in Oracle BI EE 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 victim'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 Business Intelligence Console application. When you visit this site, you are unknowingly clicking buttons on the inline-framed Console application. This vulnerability is very serious, because the attacker is not stopped by the same origin policy principles that apply to other Oracle Business Intelligence 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 Oracle BI 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 Oracle BI Presentation Services to be rendered in an inline frame.

Using the JavaHost Service for Oracle BI Presentation Services

Java library functionality can be achieved using the JavaHost service.

The JavaHost service gives Presentation Services the ability to use functionality that is 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 issues an HTTP request to another component
- Integration Service Call used by the Oracle BI Server to execute Java code



Note:

The JavaHost service uses the core libraries of BI Publisher to export the contents of analyses into various formats such as PDF, Microsoft Excel and PowerPoint, and images. BI Publisher libraries are embedded within the JavaHost service and do not depend on BI Publisher running or being deployed in a J2EE container.

In the configuration file for the JavaHost Service, elements related to the BI 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`, do not apply to the `MessageProcessor`, `Listener`, or `SSL` loaders.

Element	Description
Loaders	Contains the <code>ListOfEnabledLoaders</code> and <code>Loader</code> elements. These elements specify the components for the JavaHost service. Avoid editing the elements in the Loaders section.

Element	Description
Loaders/ListOfEnabledLoaders	<p>Specifies the list of components such as Oracle BI Scheduler and BI Publisherto enable.</p> <p>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.</p> <p>Each component has a corresponding Loader element. The name of the component listed here must match the name that is specified in the corresponding Loader/Name element.</p>
Loaders/Loader	<p>Contains the following elements, which specify configuration information for a specific component:</p> <ul style="list-style-type: none"> • 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.
InputStreamLimitInKB	<p>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.</p> <p>Default: 8192</p> <p>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 data sets. 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.</p>
RequestResponseLogDirectory	<p>A child element common to each loader that specifies the name of the directory for the response files of requests.</p> <p>Default: A default temp directory</p>
LogLargeRequests	<p>A child element common to each loader that specifies whether to create a response file when processing large requests.</p> <p>Default: true</p>
ReadRequestBeforeProcessing	<p>A child element common to each loader that specifies whether to wait to process the request until a file is completely read.</p> <p>If your organization uses the export feature in Oracle BI EE, it is 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.</p> <p>Default: true</p>

Element	Description
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: <ul style="list-style-type: none"> • 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: *
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: <ul style="list-style-type: none"> • Yes: Enables SSL encryption • No: Disables SSL encryption Default: No For information about SSL, see <i>Security Guide for Oracle Business Intelligence Enterprise Edition</i> .
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: <ul style="list-style-type: none"> • Enabled • DefaultUserJarFilePath • DefaultTempFilePath • DefaultPurgingPeriod

Element	Description
Scheduler/Enabled	<p>Specifies whether to enable the interaction of the JavaHost service with Oracle BI Scheduler to run Java jobs:</p> <ul style="list-style-type: none"> • true: Enables interaction with Oracle BI Scheduler • false: Disables interaction with Oracle BI Scheduler <p>Default: false</p>
Scheduler/DefaultUserJarFilePath	<p>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.</p> <p>When Oracle BI Scheduler is enabled, this element is required and accepts a single path.</p>
Scheduler/DefaultTempFilePath	<p>Specifies the default directory for storing temporary files for Oracle BI Scheduler requests.</p> <p>Default: the system temp directory</p>
Scheduler/DefaultPurgingPeriod	<p>Specifies the default period (in seconds) for Oracle BI Scheduler requests to remove failed jobs.</p> <p>Default: 300</p>
XMLP	<p>Contains only the common subelements such as <code>InputStreamLimitInKB</code> and <code>ReadRequestBeforeProcessing</code>, as they relate to Oracle BI Publisher.</p>
URLConnect	<p>Contains elements that relate to SSL. Avoid modifying these elements.</p>
DVT	<p>Contains only the common <code>InputStreamLimitInKB</code> child element by default, as they relate to graph generation. You can add other common child elements as necessary.</p>

C

Mapping User Interface Labels with Configuration File Elements

This appendix maps Fusion Middleware Control User Interface (UI) labels for Oracle Business Intelligence with the corresponding element names used in configuration files. .

The information in the following tables is included here for completeness. You do not need this information for most operations.

- For information about elements for the Oracle BI Server that are not 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	Using Fusion Middleware Control to Enable and Disable Query Caching
Maximum cache entry size	MAX_CACHE_ENTRY_SIZE	NQSCONFIG.INI	Using Fusion Middleware Control to Set Query Cache Parameters
Maximum cache entries	MAX_CACHE_ENTRIES	NQSCONFIG.INI	Using Fusion Middleware Control to Set Query Cache Parameters
Global cache path	GLOBAL_CACHE_STORAGE_PATH	NQSCONFIG.INI	Using Fusion Middleware Control to Set Global Cache Parameters
Global cache size	GLOBAL_CACHE_STORAGE_PATH	NQSCONFIG.INI	Using Fusion Middleware Control to Set Global Cache Parameters
Disallow RPD Updates	READ_ONLY_MODE	NQSCONFIG.INI	Using Fusion Middleware Control to Disallow RPD Updates
User Session Expiry	ClientSessionExpireMinutes	instanceconfig.xml	Using Fusion Middleware Control to Set the User Session Log-Off Period
Maximum Number of Rows Processed when Rendering a Table View	ResultRowLimit	instanceconfig.xml	Using 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	Using 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	Using 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	Using Fusion Middleware Control to Change Presentation Setting Defaults
Show section headings	ShowSectionHeadingsDefault	instanceconfig.xml	Using Fusion Middleware Control to Change Presentation Setting Defaults
Allow dashboard sections to be collapsible	CollapsibleSectionsDefault	instanceconfig.xml	Using Fusion Middleware Control to Change Presentation Setting Defaults
Pivot Tables show auto-preview	DisableAutoPreview	instanceconfig.xml	Using 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	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Port	SMTP_Port	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Display name of sender	From	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Email address of sender	Sender	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Username	mail.server	Credential found in oracle.bi.enterprise credential map	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents <i>Security Guide for Oracle Business Intelligence Enterprise Edition</i>
Password	See Username, in the preceding row.	Not Available	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Confirm password	See Username, in the preceding row.	Not Available	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents

Fusion Middleware Control UI Label	Configuration Element	Configuration File (for BI Scheduler)	Related Information
Number of retries upon failure	Try	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Maximum recipients	MaxRecipients	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Addressing method	UseBcc	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Use SSL to connect to mail server	UseSSL	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
Specify CA certificate source	This controls whether to fill in either SmtptCACertificateDir or SmtptCACertificateFile	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
CA certificate directory	SmtptCACertificateDir	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
CA certificate file	SmtptCACertificateFile	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
SSL certificate verification depth	SmtptCertificateVerificationDepth	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI Scheduler Email Settings that Affect Agents
SSL cipher list	SmtptCipherList	instanceconfig.xml	Using Fusion Middleware Control to Configure Oracle BI 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)	Using 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)	Using Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels

Fusion Middleware Control UI Label	Configuration Element	Configuration File	Related Information
Maximum File Size Note: Field in Query Logs region.	MaximumFileSizeKb	logconfig.xml (for the Presentation Services)	Using 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 the Presentation Services)	Using 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)	Using Fusion Middleware Control to Configure Log File Rotation Policy and Specify Log Levels
Error	Error	instanceconfig.xml (for Presentation Services and Scheduler) logging_config.xml (for JavaHost) ccslogging.xml (for Cluster Controller)	Using 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)	Using 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)	Using 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)	Using 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	<i>Security Guide for Oracle Business Intelligence Enterprise Edition</i>

Fusion Middleware Control UI Label	Configuration Element	Configuration File	Related Information
SSO Provider	EnabledSchemas (indirectly associated)	instanceconfig.xml	<i>Security Guide for Oracle Business Intelligence Enterprise Edition</i>

D

BI-Specific WLST Command Reference

This appendix lists BI-specific WLST commands.

For information about WLST, see [Using the WebLogic Scripting Tool \(WLST\)](#).

The following commands manage BI components. All component commands take a DOMAIN_HOME a machine name and an optional port specification

Command	Description
createOBICCSCComponent(domainHome, machine, port=None, portMonitor=None)	This command creates a new cluster component.
createOBISCHComponent(domainHome, machine, port=None, portMonitor=None, portScript=None)	This command creates a new scheduler component.
createOBIPSCComponent(domainHome, machine, port=None)	This command creates a new BI Presentation Server component.
createOBIJHComponent(domainHome, machine, port=None)	This command creates a new JavaHost component
listBISystemComponents(domainHome)	This command returns a list of all the system components in the domain.
getBISystemComponents(domainHome, instanceId)	This command displays details of system component with specified instanceID.
deleteBISystemComponent(domainHome, instanceId)	This command deletes a system component with specified instanceID.

The following commands manage the domain service instance.

Command	Description
listBIServiceInstances	This command lists all Service Instance keys in the BI domain.
getBIServiceInstance	This command gets service instance details for a given service instance key.
scaleOutBIServiceInstance	This command scales-out the Service Instance(s) onto the new computer, ensuring that the service instance is available on the specified computer within the BI domain. This command is only used in advanced cases.
exportServiceInstance	This command exports a service instance to a given export directory in the form of BAR file.
importServiceInstance	This command imports an already exported bar (service instance) as a customization to a given environment.
refreshServiceInstance	This command refreshes certain aspects of a service instance service Key that are inherited from the BI domain. For example, if a new product is added to the BI domain (for example, Calculation Manager template is deployed post domain creation), the permission sets for that product will only be made available to the service instance when the service instance is refreshed.

Command	Description
refreshDomainServiceInstances	This command refreshes all the service instances of the domain. The equivalent to calling refreshServiceInstance on each service instance.
resetServiceInstance	This command resets the given service instance to empty state equivalent to importing the empty BAR file.

The `sync_midtier_db` command synchronizes connection details with the mid-tier database.

Command	Description
<code>sync_midtier_db.sh cmd</code>	<p>This command synchronizes connection details to the mid-tier database ensuring that BI components can access the mid-tier database when connection details (including credentials) are changed.</p> <p>Location of command: <code>DOMAIN_HOME/bitools/bin/sync_midtier_db.sh cmd</code></p> <p>Supported mid-tier database types are DB2, SQLServer, MSSQL, and Oracle.</p> <p>On UNIX this must be performed on the master host. On Windows this must be performed on every host.</p> <p>Supported mid-tier datasources must be created by RCU, and management is limited to the following component id: BIPLATFORM — There is no validation, you must validate connectivity after updating Datasource connection pools in Weblogic Admin console or WLST before synchronizing.</p> <p>To Synchronize mid-tier database connection details:</p> <ol style="list-style-type: none"> 1. Execute synchronisation script: <code>DOMAIN_HOME/bitools/bin/sync_midtier_db.sh cmd</code>. 2. Script displays Datasources that were updated. 3. Re-start Managed Server and BI System Components. <p>For example, in <code>DOMAIN_HOME/bitools/bin/</code> enter: <code>./start.sh</code>.</p>

The commands for cloning and deleting a Oracle BI EE machine are:

- `cloneBIMachine(domainHome, listenAddress,baseMachine=None, baseServer='bi_server1', machineName=None)`
- `deleteBIMachine(domainHome, machine)`