

**Oracle® Business Intelligence Applications**

Installation Guide for Informatica PowerCenter Users

Version 7.9.5.1

**E13765-01**

January 2009

E13765-01

Copyright © 2009, Oracle. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

---

---

# Contents

<b>Preface</b> .....	xi
Audience .....	xi
Documentation Accessibility .....	xii
Related Documents .....	xii
Conventions .....	xii
<b>Part I Getting Started</b>	
<b>1 What's New in This Release</b>	
1.1 What's New in <i>Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users?</i> .....	1-1
<b>2 Overview of Oracle BI Applications</b>	
2.1 What Is Oracle BI Applications? .....	2-1
2.2 Oracle Business Analytics Warehouse Overview .....	2-2
2.3 Oracle Business Analytics Warehouse Architecture.....	2-2
2.3.1 Oracle Business Analytics Warehouse Architecture Components.....	2-2
2.4 Roadmap to Installing and Setting Up Oracle BI Applications .....	2-4
2.5 Using Oracle BI Repository Documentation.....	2-4
<b>Part II Installation and Set Up Steps for Oracle BI Applications</b>	
<b>3 Preinstallation and Predeployment Requirements for Oracle BI Applications</b>	
3.1 Operating System, Driver, and Connectivity Requirements for Oracle Business Analytics Warehouse .....	3-2
3.2 General Guidelines for Setting Up Oracle Business Analytics Warehouse.....	3-2
3.3 IBM DB2 UDB-Specific Database Guidelines for Oracle Business Analytics Warehouse	3-4
3.4 IBM DB2 UDB zOS and OS/390 and z/OS-Specific Database Guidelines for Oracle Business Analytics Warehouse .....	3-6
3.5 SQL Server-Specific Database Guidelines for Oracle Business Analytics Warehouse .....	3-6
3.5.1 Setting the ANSI NULL Option.....	3-7
3.5.2 Modifying the DB Library Options Setting.....	3-7
3.5.3 Recommended SQL Server Database Parameters .....	3-7
3.6 Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse .....	3-9
3.6.1 Installation of Required JDBC Driver for Teradata Databases .....	3-9

3.6.2	General Guidelines for Teradata Deployments.....	3-9
3.6.3	Best Practices For Teradata Deployments.....	3-9
3.6.3.1	LEFT OUTER JOIN Issue .....	3-10
3.6.3.2	Additional Statistics Required .....	3-11
3.6.3.2.1	Collect Statistics As A Prerequisite.....	3-11
3.6.3.2.2	Group By Vs Distinct .....	3-11
3.6.3.2.3	Pruning Of Mapping.....	3-11
3.6.3.2.4	Loader Configurations.....	3-11
3.6.3.2.5	Fastload .....	3-12
3.6.3.2.6	Mload .....	3-12
3.6.3.2.7	Tpump.....	3-12
3.7	Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse.....	3-13
3.8	Additional Suggestions for Optimizing Oracle Performance in Oracle Business Analytics Warehouse .....	3-13
3.9	Miscellaneous Information About Oracle BI Applications Deployments.....	3-14
3.9.1	Preconfigured Mapping for the SA System Subject Area.....	3-15
3.9.2	Using Initialization Blocks.....	3-16
3.9.3	Creating Custom Indexes in Siebel Source Databases for Incremental Load Performance .....	3-16
3.9.3.1	An Example of Change Capture SQL and Required Indexes .....	3-17
3.9.4	Creating Custom Indexes in Oracle EBS Source Database for Incremental Load Performance.....	3-17
3.9.4.1	Creating Indexes for Category 1 Tables .....	3-18
3.9.4.2	Creating Indexes for Category 2 Tables .....	3-19
3.9.5	Running an ETL with a Non-English OLTP Data Source.....	3-20
3.10	Partitioning Guidelines For Large Fact Tables .....	3-21
3.10.1	Introduction to Partitioning Large Fact Tables .....	3-21
3.10.2	How to Partition Large Fact Tables.....	3-21
3.10.3	Configure DAC to Support ETL for Partitioned Tables.....	3-24
3.10.4	Example XML Files.....	3-27
3.10.4.1	Pre_W_REVN_F_ini.xml - Pre SQL XML File Name for Full Load .....	3-28
3.10.4.2	Post_W_REVN_F_ini.xml - Post SQL XML File Name for Full Load.....	3-28
3.10.4.3	Pre_W_REVN_F_incr.xml- Pre SQL XML File Name for Incremental Load....	3-28
3.10.4.4	Post_W_REVN_F_incr.xml- Post SQL XML File Name for Incremental Load	3-29

## 4 Installing and Setting Up Oracle BI Applications

4.1	About Oracle BI Applications Topologies.....	4-2
4.2	High-Level Installation and Configuration Task List.....	4-3
4.3	Mandatory Requirements and Preinstallation Tasks .....	4-5
4.3.1	Oracle Business Intelligence Infrastructure Requirements.....	4-5
4.3.2	Informatica PowerCenter Requirements.....	4-6
4.3.3	Code Page and Data Movement Requirements .....	4-6
4.3.4	Preinstallation Tasks .....	4-8
4.3.4.1	Create Databases for Oracle BI Applications and Informatica PowerCenter Components .....	4-8
4.3.4.1.1	How to Create the SSE Role.....	4-9
4.3.4.2	Install Database Connectivity Software .....	4-9
4.3.4.2.1	How to Set the NLS_LANG Environment Variable for Oracle Databases	4-10

4.3.4.2.2	How to Set the DB2CODEPAGE Environment Variable for DB2 Databases .....	4-11
4.3.4.3	Create ODBC Database Connections.....	4-12
4.3.4.3.1	How to Create ODBC Connections for DB2 Databases.....	4-12
4.3.4.3.2	How to Create ODBC Connections for Oracle Databases.....	4-12
4.3.4.3.3	How to Create ODBC Connections for SQL Server Databases .....	4-13
4.3.4.3.4	How to Create ODBC Connections for Teradata Databases.....	4-13
4.3.4.4	Set Environment Variables.....	4-14
4.3.4.5	Download and Install the Java JDK.....	4-14
4.3.4.6	Download Hibernate Libraries.....	4-15
4.4	Running the Oracle BI Applications Installer on Windows .....	4-15
4.5	Installing Informatica PowerCenter Services Software.....	4-17
4.5.1	Installing Informatica PowerCenter Client Tools (Windows) .....	4-18
4.5.2	Installing Informatica PowerCenter Services .....	4-18
4.6	Restoring the Prebuilt Informatica Repository.....	4-21
4.7	Configuring Informatica PowerCenter Components .....	4-23
4.7.1	Copying Source Files and Lookup Files.....	4-23
4.7.2	Setting PowerCenter Integration Services Relaxed Code Page Validation.....	4-23
4.7.3	Setting PowerCenter Integration Services Custom Properties .....	4-24
4.8	Installing an Additional Instance of the DAC Client on Windows.....	4-25
4.9	Configuring the DAC Client .....	4-25
4.9.1	Copying Hibernate Files to the DAC Directories.....	4-26
4.9.2	Installing JDBC Drivers for DAC Database Connectivity .....	4-26
4.9.3	Configuring the DAC Config.bat File.....	4-27
4.9.4	Enabling DAC Client Communication with Informatica PowerCenter.....	4-27
4.9.4.1	How to Install Informatica pmcmd and pmrep Command Line Programs.....	4-27
4.9.4.2	How to Define the Informatica Domains File Path in the DAC Client Environment Variables .....	4-28
4.9.4.3	How to Verify the DAC Client Is Able to Use pmrep and pmcmd .....	4-29
4.10	Logging into DAC for the First Time and Creating the DAC Repository .....	4-29
4.10.1	Logging into DAC .....	4-29
4.10.2	Creating the DAC Repository Schema .....	4-31
4.10.3	Importing Metadata into the DAC Repository.....	4-31
4.11	Creating the Oracle Business Analytics Warehouse Tables .....	4-33
4.11.1	Creating Data Warehouse Tables.....	4-34
4.11.2	Creating Data Warehouse Tables on a Teradata Database.....	4-37
4.12	Installing the DAC Server.....	4-38
4.12.1	Installing an Additional Instance of the DAC Server on Windows.....	4-38
4.12.2	Installing the DAC Server on UNIX.....	4-39
4.13	Configuring the DAC Server.....	4-41
4.13.1	Configuring the Connection Between the DAC Server and DAC Repository .....	4-41
4.13.1.1	How to Configure the DAC Server Repository Connection Using the DAC Client (Windows) .....	4-42
4.13.1.2	How to Configure the DAC Server Repository Connection Using serverSetupPrompt Scripts (Windows, UNIX or Linux).....	4-44
4.13.2	Enabling DAC Server Communication with Informatica PowerCenter .....	4-44
4.13.2.1	How to Set Environment Variables on Windows.....	4-45

4.13.2.2	How to Set Environment Variables on UNIX or Linux .....	4-45
4.13.2.3	How to Verify Java JDK Availability and Version .....	4-47
4.13.3	Starting the DAC Server .....	4-47
4.13.4	Activating Join Indexes for Teradata Databases .....	4-47
4.14	Configuring DAC Integration Settings .....	4-48
4.14.1	Setting DAC System Properties .....	4-48
4.14.2	Registering Informatica Services in DAC.....	4-49
4.14.3	Setting Physical Data Sources .....	4-51
4.15	Configuring Relational Connections in Informatica Workflow Manager.....	4-53
4.15.1	Logging Into Informatica PowerCenter Workflow Manager.....	4-53
4.15.2	Configuring Relational Connections in Informatica PowerCenter Workflow Manager .....	4-54
4.15.2.1	How to Configure Relational Connections.....	4-54
4.15.2.2	(Teradata specific) How to Configure the Informatica Repository for Teradata External Loader Connections.....	4-56
4.15.2.2.1	How to Specify Loader Connections for Teradata .....	4-56
4.15.2.2.2	How to Specify Teradata Details at the Workflow Level .....	4-56
4.16	Configuring the SiebelUnicodeDB Custom Property.....	4-58
4.17	Setting Up DAC to Receive Email Notification .....	4-59
4.17.1	Configuring Email Recipients in DAC .....	4-59
4.17.2	Configuring Email in the DAC Server.....	4-60
4.18	Additional Configuration Tasks .....	4-60
4.18.1	Setting DAC Source System Parameters .....	4-61
4.18.2	Creating Stored Procedures for DB2-UDB.....	4-62
4.18.2.1	How to Verify the Existence of Stored Procedures.....	4-62
4.18.2.2	How to Create DB2 Stored Procedures .....	4-62
4.18.3	Siebel-Specific Configuration Tasks.....	4-63
4.18.3.1	Migrating Seed Data from a Siebel Source System to the Oracle BI Repository .....	4-63
4.18.3.2	Updating Siebel Transactional Database Schema Definitions .....	4-64
4.18.3.2.1	How to Run the DDLIMP Tool From The Command Line .....	4-64
4.18.3.2.2	How to Apply Siebel CRM Schema Changes for Oracle, DB2/UDB, MSSQL .....	4-64
4.18.3.2.3	How to Apply CRM Schema Changes for DB2 on OS/390 and z/OS.....	4-65
4.18.3.2.4	How to Verify the Siebel CRM Schema Changes .....	4-66
4.18.3.2.5	About Delete Triggers .....	4-66
4.18.4	Teradata-Specific Configuration Tasks.....	4-67
4.18.4.1	Setting Up the Code Page File Property for Unicode Environments on Teradata Databases .....	4-67
4.18.4.2	Setting Up the HOSTS File for Teradata Installations on Windows.....	4-67
4.18.4.3	Setting Up the HOSTS File for Teradata Installations on UNIX.....	4-68
4.18.4.4	Teradata-Specific Installation Checklist .....	4-68
4.19	About Running A Full Load ETL .....	4-69
4.19.1	An Example of Running a Full Load ETL.....	4-69

## Part III Appendices

## **A Supporting Tasks for DAC and Informatica PowerCenter**

A.1	How to Log into DAC .....	A-1
A.2	How to Set DAC System Properties .....	A-2
A.3	How to Start and Stop the DAC Server .....	A-4
A.4	How to Log Into Informatica PowerCenter Administration Console.....	A-6
A.5	How to Start and Stop Informatica Services .....	A-7
A.6	How to Stop and Start the Informatica Repository Service (Optional).....	A-7
A.7	About the DAC Server Scripts for Bash and C Shells.....	A-8
A.8	How to Configure the DAC Metadata Repository .....	A-9
A.8.1	How to Set Up DAC System Properties .....	A-9
A.8.2	How to Activate Join Indices for Teradata Databases.....	A-13

## **B Configuring Universal Adapter for CRM Analytics**

B.1	What is Universal Adapter for CRM Analytics? .....	B-1
B.2	Data Format Requirements.....	B-1
B.3	About the Sample Source Files Provided .....	B-2
B.4	How to Create Data Files for Universal Adapter for CRM Analytics.....	B-2
B.5	List of Sample Source Files for Universal Adapter for CRM Analytics.....	B-2
B.5.1	List of Source Files for Seed Data and Common Dimensions.....	B-3
B.5.2	List of Source Files for Specific Subject Areas .....	B-3

## **C Localizing Oracle Business Intelligence Deployments**

C.1	Process of Maintaining Translation Tables for Oracle BI.....	C-1
C.1.1	Upgrading Oracle Business Intelligence Seed Data for Non-English Locales.....	C-2
C.1.1.1	Importing Locale Seed Data Into The Translation Table (W_LOCALIZED_STRING_G).....	C-2
C.1.2	Externalizing Customer Metadata Strings .....	C-3
C.1.3	Adding Custom Translations to the W_LOCALIZED_STRING_G Table .....	C-4
C.1.3.1	Adding String Translations for Analytics Metadata .....	C-4
C.2	About Translating Presentation Services Strings .....	C-5
C.3	Changing the Default Currency in Analytics Applications.....	C-5

## **D Integrating Interactive Dashboards and Operational Applications Data**

D.1	Importing Oracle's Siebel Industry Applications Seed Data .....	D-1
D.2	Completing the Initialization in the Siebel Operational Application.....	D-2
D.2.1	Updating the Siebel Operational Application .....	D-2
D.2.2	Reapplying Customized Style Sheets .....	D-3
D.2.3	How to Configure Oracle Business Intelligence with Oracle's Siebel Web Extension (SWE) .....	D-4
D.2.4	Creating a Virtual IP Address for the SWE and Oracle BI Presentation Services.....	D-5
D.2.5	Testing the Virtual IP Configuration for Oracle Business Intelligence and the SWE .....	D-5
D.2.6	How to Configure Oracle BI Action Links.....	D-6
D.2.7	Testing Action Links .....	D-6
D.2.8	Process of Integrating Analytics and Oracle's Siebel Operational Applications.....	D-6
D.2.9	Accessing Optional Analytics Applications .....	D-7

D.2.10	Configuring the Browser Settings for Operational Applications .....	D-7
D.2.11	Process of Customizing Oracle BI Application Home Page and Dashboards.....	D-8
D.2.12	Customizing Oracle BI Content on the Siebel Operational Application Home Page.....	D-8
D.2.13	Determining the Oracle BI Report Path Argument .....	D-10
D.2.14	Adding Views for Custom Oracle BI Interactive Dashboards .....	D-11
D.2.15	Verifying NQHOST Setup for Pharma Disconnected Analytics Client .....	D-11
D.2.16	Integrating the Pharma Application Home Page with the Pharma Disconnected Analytics Client.....	D-12
D.2.17	Importing Pharma Application Target Lists to the Pharma Disconnected Analytics Client .....	D-13
D.3	Viewing Information About Preconfigured Dashboards .....	D-14
D.3.1	Exposing Dashboards and Requests.....	D-14
D.3.2	Locating Dashboard and Page Names.....	D-14

## **E Configuring Metadata for Analytics Applications**

E.1	What Are Oracle's Siebel Operational Applications? .....	E-1
E.2	Updating Financial Analytics Logical Table Sources .....	E-2
E.3	Externalizing Financial Services Metadata Translation Strings.....	E-3
E.4	Disabling the Financial Analytics Logical Table Sources.....	E-3
E.5	Developing and Deploying Predictive Scores .....	E-4
E.6	Analytics Metadata Requirements for Oracle's Siebel Industry Applications.....	E-5
E.6.1	Oracle's Siebel Consumer Sector Dashboards and Pages .....	E-5
E.6.2	Oracle's Siebel Consumer Sector Data Requirements .....	E-7
E.6.3	Oracle's CME family of products (Oracle Communications, Media and Energy Sales Analytics.....	E-7
E.6.4	Oracle's Siebel CME Dashboards and Pages .....	E-9
E.6.5	Oracle's Siebel CME Data Requirements.....	E-10
E.6.6	Oracle Pharma Sales Analytics Dimensions .....	E-10
E.6.6.1	Positions Dimension.....	E-10
E.6.6.2	Alignments Dimension.....	E-10
E.6.6.3	Products Dimension.....	E-11
E.6.6.4	Product Costs Dimension.....	E-11
E.6.7	Dimensions Specific to Subject Areas in Oracle Pharma Sales Analytics and Oracle Pharma Marketing Analytics .....	E-12
E.6.7.1	Pharma Sales Effectiveness .....	E-12
E.6.7.2	Pharma Product Categories .....	E-12
E.6.7.3	Pharma Promotional Effectiveness .....	E-12
E.6.7.4	Pharma Medical Education Effectiveness.....	E-13
E.6.7.5	Pharma Objectives Achievement` .....	E-13

## **F Oracle Business Analytics Warehouse for Life Sciences Data Considerations**

F.1	What's New for Pharma Analytics in Version 7.9.....	F-1
F.2	Oracle Business Analytics Warehouse for Life Sciences Overview.....	F-1
F.3	Importing Syndicated Data into Oracle Business Analytics Warehouse.....	F-2
F.3.1	Syndicated Loading Definitions .....	F-3
F.3.2	Data Types Supported in the Oracle Business Analytics Warehouse.....	F-3



F.3.3	Loading Syndicated Data into the Oracle Business Analytics Warehouse .....	F-4
F.3.3.1	Loading From a Flat File Source Using Syndicated Data Staging Tables and Cross-Referencing Tables .....	F-4
F.3.3.1.1	Flat File Data Population Rules for Populating TRx Data.....	F-6
F.3.3.2	Loading From a Flat File Source Using Syndicated Market Staging Tables.....	F-9
F.4	Syndicated Data Flat File Formats.....	F-10
F.4.1	Syndicated Data Flat File for Direct Sales Data.....	F-10
F.4.2	Syndicated Data Flat File for Indirect Sales Data.....	F-11
F.4.3	Syndicated Data Flat File for Prescription Data .....	F-13
F.4.4	Syndicated Data Flat File for Plan Level Rx Data .....	F-14
F.4.5	Syndicated Data Flat File for Physician Plan Level Rx Data .....	F-16
F.4.6	Syndicated Data Flat File for Weekly Early View Data .....	F-17
F.4.7	Syndicated Market Flat File for Indirect Sales Data .....	F-19
F.4.8	Syndicated Market Flat File for Prescription Data.....	F-20
F.4.9	Syndicated Market Flat File for Plan Level Rx Data.....	F-22
F.4.10	Syndicated Market Flat File for Physician Plan Level Rx Data.....	F-23
F.4.11	Syndicated Market Flat File for Weekly Early View Data .....	F-25
F.4.12	Flat File Cross-Reference Data Population Rules.....	F-26
F.5	Life Sciences Data Loading Issues with Oracle Business Analytics Warehouse .....	F-30
F.5.1	Known Issues with the Syndicated Data Fact Tables .....	F-30
F.6	Incremental Updates in the Oracle Business Analytics Warehouse LS Dimension Tables.....	F-31
F.6.1	W_ALIGNMT_DH .....	F-31
F.6.2	W_ALIGNVER_DH.....	F-31

## **G Pharma Disconnected Analytics Administration - Supplemental Information**

G.1	About Pharma Disconnected Analytics.....	G-1
G.2	Sourcing Reports for Oracle's Pharma Disconnected Analytics .....	G-2

## **H Using Oracle Business Analytics Warehouse Exception Reports**

H.1	Understanding Oracle Business Analytics Warehouse Exceptions.....	H-1
H.1.1	List of Values Exceptions.....	H-1
H.1.2	Cost List Exceptions .....	H-2
H.1.3	Products Without a Cost List .....	H-3
H.1.4	Exchange Rate Exceptions .....	H-3
H.1.5	Invalid Hierarchy Exceptions .....	H-3
H.1.6	Circular Hierarchy Exceptions.....	H-3
H.2	Executing Oracle Business Analytics Warehouse Exception Reports.....	H-4
H.2.1	Cleansing Data .....	H-4
H.2.2	Using the List of Values View.....	H-5
H.2.3	Using the Exchange Rates View .....	H-5
H.2.4	Using the Cost List View .....	H-6
H.2.5	Using the ETL History View .....	H-6
H.2.6	Additional Exceptions.....	H-6

## **I Troubleshooting Oracle BI Applications**

1.1	Oracle Business Analytics Warehouse Installation and Initial Configuration Issues.....	I-2
1.2	Informatica and Loading Issues.....	I-2
1.3	Unicode Issues with Informatica and the Oracle Business Analytics Warehouse .....	I-6
1.4	UNIX Installation Issues with the Oracle Business Analytics Warehouse.....	I-7
1.5	Oracle Business Analytics Warehouse Upgrade Issues .....	I-8
1.6	Oracle BI Interactive Dashboard Issues.....	I-9
1.7	ETL Issues .....	I-10

## **J About the Versioned Informatica Repository**

J.1	Summary of Versioning in the Informatica Repository .....	J-1
J.2	Customization of Repository Objects Using Check Out and Check In.....	J-1

## **Index**

---

---

# Preface

Oracle Business Intelligence Applications are comprehensive prebuilt solutions that deliver pervasive intelligence across an organization, empowering users at all levels — from front line operational users to senior management — with the key information they need to maximize effectiveness. Intuitive and role-based, these solutions transform and integrate data from a range of enterprise sources, including Siebel, Oracle, PeopleSoft, and corporate data warehouses — into actionable insight that enables more effective actions, decisions, and processes.

Oracle BI Applications are built on Oracle Business Intelligence Suite Enterprise Edition, a comprehensive next-generation BI and analytics platform.

Oracle BI Applications includes the following application families:

- Sales
- Service and Contact Center
- Marketing
- Financial
- Supply Chain and Supplier
- HR/Workforce

*Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users* contains instructions for installing and setting up Oracle BI Applications release 7.9.5.1. For a high-level road map of the steps required to install and set up Oracle BI Applications, see [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications."](#)

Oracle recommends reading the *Oracle Business Intelligence Applications Release Notes* before installing, using, or upgrading Oracle BI Applications. The *Oracle Business Intelligence Applications Release Notes* are available:

- On the Oracle Business Intelligence Applications CD-ROM.
- On the Oracle Technology Network at [http://www.oracle.com/technology/documentation/bi\\_apps.html](http://www.oracle.com/technology/documentation/bi_apps.html) (to register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technology/about/index.html>).

## Audience

This document is intended for BI managers and implementers of Oracle BI Applications.

## Documentation Accessibility

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

### Accessibility of Code Examples in Documentation

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

### Accessibility of Links to External Web Sites in Documentation

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

### TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

## Related Documents

For more information, see the following documents in the Oracle BI Applications release 7.9.5.1 documentation set (available at [http://www.oracle.com/technology/documentation/bi\\_apps.html](http://www.oracle.com/technology/documentation/bi_apps.html)):

- *Oracle Business Intelligence Applications Release Notes*
- *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*
- *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*
- *Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users*
- *Oracle Business Intelligence Applications Security Guide*
- *Oracle Business Intelligence Data Warehouse Administration Console Guide*
- *Oracle Business Analytics Warehouse Data Model Reference*

## Conventions

The following text conventions are used in this document:

<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	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.



# Part I

---

## Getting Started

Part I helps you get started with Oracle BI Applications. It contains the following sections:

- [Chapter 1, "What's New in This Release"](#)
- [Chapter 2, "Overview of Oracle BI Applications"](#)

---

**Note:** For a high-level road map for installation and set up steps for Oracle BI Applications, see [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications."](#)

---





---

---

## What's New in This Release

This section lists changes described in this version of the documentation to support release 7.9.5.1 of the software.

### 1.1 What's New in *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*?

The *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users* contains installation and set up information that previously appeared in *Oracle Business Intelligence Applications Installation and Configuration Guide*.

Information about configuration and customization that previously appeared in *Oracle Business Intelligence Applications Installation and Configuration Guide* now appears in *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*.

Information about implementing security with Oracle BI Applications that previously appeared in *Oracle Business Intelligence Applications Installation and Configuration Guide* now appears in *Oracle Business Intelligence Applications Security Guide*.

For this release, quality enhancements were made to many sections in *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. This guide also includes the following new and changed topics:

- [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications,"](#) was updated to reflect the reorganization of information.
- [Section 3.6.1, "Installation of Required JDBC Driver for Teradata Databases"](#) was added as a new topic.
- [Section 4.4, "Running the Oracle BI Applications Installer on Windows,"](#) was updated to reflect installation instructions for release 7.9.5.1.
- [Section 4.5, "Installing Informatica PowerCenter Services Software,"](#) was updated to reflect the requirement to install Informatica PowerCenter version 8.1.1 SP5.



---



---

## Overview of Oracle BI Applications

This chapter provides an overview of Oracle BI Applications, and includes the following topics:

- [Section 2.1, "What Is Oracle BI Applications?"](#)
- [Section 2.2, "Oracle Business Analytics Warehouse Overview"](#)
- [Section 2.3, "Oracle Business Analytics Warehouse Architecture"](#)
- [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications"](#)
- [Section 2.5, "Using Oracle BI Repository Documentation"](#)

### 2.1 What Is Oracle BI Applications?

Oracle BI Applications is a prebuilt business intelligence solution.

Oracle BI Applications supports Oracle sources, such as Oracle E-Business Suite Applications, Oracle's Siebel Applications, Oracle's PeopleSoft Applications, and non-Oracle sources, such as SAP Applications. If you already own one of the above applications, you can purchase Oracle Business Intelligence Enterprise Edition and Oracle BI Applications to work with the application.

Oracle BI Applications also provides complete support for enterprise data, including financial supply chain and workforce sources. These enterprise applications typically source from both Oracle data sources, such as Oracle EBS and PeopleSoft and non-Oracle data sources, such as SAP.

Oracle BI Applications consists of the components shown in [Table 2-1](#).

**Table 2-1 Oracle BI Applications Components (with Informatica/DAC)**

Component	Description
DAC Repository metadata files	This content includes repository objects such as tables, subject areas, execution plans, and tasks, and is contained in XML files.
Embedded Informatica ETL tool	This is a third-party application that performs the extract, transform, and load (ETL) operations for the data warehouse.
Prebuilt Informatica content	This content includes ETL repository objects, such as mappings, sessions, and workflows, and is contained in the Informatica Repository file (Oracle_BI_DW_Base.rep).
Prebuilt metadata content	This metadata content is contained in the Oracle BI Applications repository file (OracleBIAnalyticsApps.rpd).

**Table 2–1 (Cont.) Oracle BI Applications Components (with Informatica/DAC)**

Component	Description
Prebuilt reports and dashboard content	This content is contained in the Oracle BI Presentation Services Catalog.
Oracle Business Analytics Warehouse	The prebuilt data warehouse that holds data extracted, transformed, and loaded from the transactional database. For more information, see <a href="#">Section 2.2, "Oracle Business Analytics Warehouse Overview."</a>

## 2.2 Oracle Business Analytics Warehouse Overview

The Oracle Business Analytics Warehouse is a unified data repository for all customer-centric data, which supports the analytical requirements of the supported source systems.

The Oracle Business Analytics Warehouse includes the following:

- A complete relational enterprise data warehouse data model with numerous prebuilt star schemas encompassing many conformed dimensions and several hundred fact tables.  
  
For more information about the data warehouse data model, see the *Oracle Business Analytics Warehouse Data Model Reference*.
- An open architecture to allow organizations to use third-party analytical tools in conjunction with the Oracle Business Analytics Warehouse using the Oracle Business Intelligence Server.
- Prebuilt data extractors to incorporate data from external applications into the Oracle Business Analytics Warehouse.
- A set of ETL (extract-transform-load) processes that takes data from a wide range of source systems and creates the Oracle Business Analytics Warehouse tables.
- The Oracle Business Intelligence Data Warehouse Administration Console (DAC), a centralized console for the set up, configuration, administration, loading, and monitoring of the Oracle Business Analytics Warehouse.

## 2.3 Oracle Business Analytics Warehouse Architecture

High-level analytical queries, like those commonly used in Oracle Business Intelligence, scan and analyze large volumes of data using complex formulas. This process can take a long time when querying a transactional database, which impacts overall system performance.

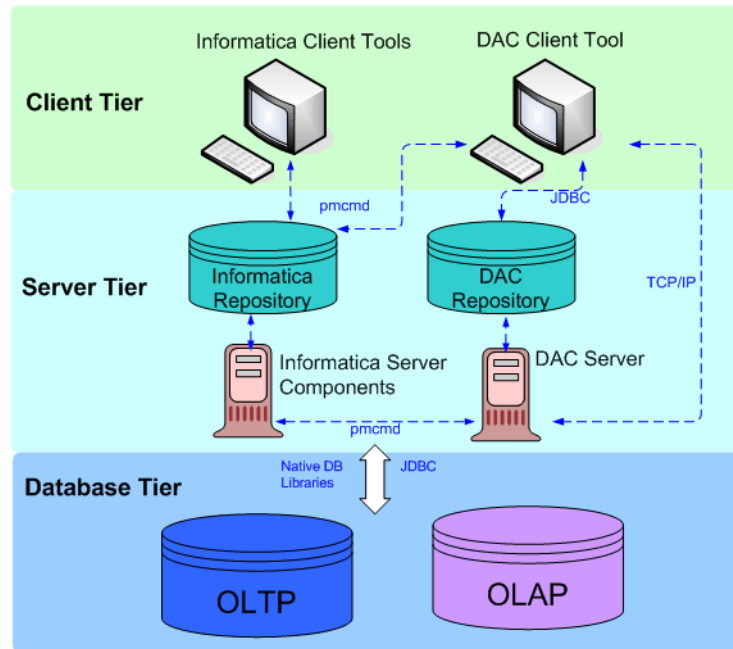
For this reason, the Oracle Business Analytics Warehouse was constructed using dimensional modeling techniques to allow for fast access to information required for decision making. The Oracle Business Analytics Warehouse derives its data from operational applications and uses Informatica PowerCenter to extract, transform, and load data from various supported transactional database systems (OLTP) into the Oracle Business Analytics Warehouse.

### 2.3.1 Oracle Business Analytics Warehouse Architecture Components

The figure above shows the following Oracle Business Analytics Warehouse components:

Figure 2–1 illustrates the Oracle Business Analytics Warehouse architecture when deployed with Informatica PowerCenter and DAC.

**Figure 2–1 An example Oracle Business Analytics Warehouse architecture with Informatica PowerCenter and DAC**



- The Client tier contains the Informatica PowerCenter client tools and the Oracle BI Data Warehouse Administration Console (DAC). DAC is a command and control interface for the data warehouse to allow for set up, configuration, administration, and monitoring of data warehouse processes.
- The Server tier contains the following:
  - DAC Server. Executes the instructions from the DAC Client. The DAC Server manages data warehouse processes, including scheduling, loading of the ETL, and configuring the subject areas to be loaded. It dynamically adjusts its actions based on information in the DAC Repository. Depending on your business needs, you might incrementally refresh the Oracle Business Analytics Warehouse once a day, once a week, once a month, or on another similar schedule.
  - DAC Repository. Stores the metadata (semantics of the Oracle Business Analytics Warehouse) that represents the data warehouse processes.
  - Informatica PowerCenter Services:
    - Integration Services - The Integration Service reads workflow information from the repository. The Integration Service connects to the repository through the Repository Service to fetch metadata from the repository.
    - Repository Services - The Repository Service manages connections to the PowerCenter Repository from client applications. The Repository Service is a separate, multi-threaded process that retrieves, inserts, and updates metadata in the repository database tables.
  - Informatica Repository. Stores the metadata related to Informatica workflows.
- The Database tier contains the OLTP and OLAP databases.

## 2.4 Roadmap to Installing and Setting Up Oracle BI Applications

To install and set up Oracle BI Applications, do the following:

- For the source systems that you are using, follow the appropriate preinstallation steps in [Chapter 3, "Preinstallation and Predeployment Requirements for Oracle BI Applications."](#)
- Install and set up the Oracle BI Applications components and Informatica PowerCenter components as described in [Chapter 4, "Installing and Setting Up Oracle BI Applications."](#)

**Note:** For an example that shows a typical deployment topology for Oracle BI Applications, see [Section 4.1, "About Oracle BI Applications Topologies."](#)

- Perform the source-independent steps that are required before a full data load, described in the section entitled, "Configuring Common Areas and Dimensions," in *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*. Then, perform the source system-specific steps that are required before a full data load for the appropriate source system that you are using.
- Perform any required additional source-independent steps described in the section entitled, "Configuration Steps for Controlling Your Data Set for All Source System," in *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*. Then, perform any required additional steps for the appropriate source system that you are using.
- Perform the configuration steps that are required before a full data load for all applications that you are deploying. See *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users* for instructions on configuring the different application families.
- (Optional) If you want to customize the preconfigured Oracle BI Applications functionality, follow the steps described in the chapter entitled, "Customizing the Oracle Business Analytics Warehouse," in the *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*.
- (Optional) If you want to modify the preconfigured Oracle BI Applications security, see *Oracle Business Applications Security Guide*.

Once you have installed and set up Oracle BI Applications components, configured the modules (optional), and customized Oracle BI Applications (optional), you are ready to start running ETL processes.

For a detailed example of how to run an ETL for Oracle Financials with an Oracle EBS OLTP data source, see [Section 4.19, "About Running A Full Load ETL"](#).

For information about running ETL processes, see *Oracle Business Intelligence Data Warehouse Administration Console Guide*.

## 2.5 Using Oracle BI Repository Documentation

When you deploy Oracle BI Applications, you can use the following documentation and tools to manage your metadata:

- Oracle BI Repository Documentation

Using Oracle BI Administration Tool, you can generate repository documentation that lists the mapping from the presentation columns to the corresponding logical and physical columns. You might use this information for gap-analysis, or to create a record of your repository that you can use to compare with other repositories.

To generate repository documentation into a text or comma-separated file, log into Oracle BI Administration Tool and choose Tools, then Utilities, then Repository Documentation.

For more information about generating repository documentation, see *Oracle Business Intelligence Server Administration Guide*.

- Presentation Services Catalog

Using the Catalog Manager, you can view the names of the prebuilt dashboards and requests in the Presentation Services Catalog.

To view the Presentation Catalog, select Catalog Manager from the Windows \ Start \ Programs \ Oracle Business Intelligence menu.





# Part II

---

## Installation and Set Up Steps for Oracle BI Applications

---

Part II provides instructions for installing and setting up Oracle BI Applications. It contains the following sections:

- [Chapter 3, "Preinstallation and Predeployment Requirements for Oracle BI Applications"](#)
- [Chapter 4, "Installing and Setting Up Oracle BI Applications"](#)

---

**Note:** For a high-level road map for installation, configuration, and customization steps for Oracle BI Applications, see [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications."](#)

---



---

---

## Preinstallation and Predeployment Requirements for Oracle BI Applications

This section provides information about preparing to install and deploy Oracle BI Applications. You should review this information before you begin the installation and deployment process. You should also read the general guidelines for setting up the Oracle Business Analytics Warehouse and read the appropriate database-specific guidelines for the source OLTP databases that you are using.

You also need to satisfy the database and Informatica PowerCenter requirements that are specified in [Chapter 4, "Installing and Setting Up Oracle BI Applications."](#) For more information, see [Section 4.3, "Mandatory Requirements and Preinstallation Tasks."](#)

### Notes

- For information about database-specific settings, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.
- For information about code page settings, refer to the Informatica documentation.

This section contains the following topics:

- [Section 3.1, "Operating System, Driver, and Connectivity Requirements for Oracle Business Analytics Warehouse"](#)
- [Section 3.2, "General Guidelines for Setting Up Oracle Business Analytics Warehouse"](#)
- [Section 3.3, "IBM DB2 UDB-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)
- [Section 3.4, "IBM DB2 UDB zOS and OS/390 and z/OS-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)
- [Section 3.5, "SQL Server-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)
- [Section 3.6, "Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)
- [Section 3.7, "Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)
- [Section 3.8, "Additional Suggestions for Optimizing Oracle Performance in Oracle Business Analytics Warehouse"](#)
- [Section 3.9, "Miscellaneous Information About Oracle BI Applications Deployments"](#)
- [Section 3.10, "Partitioning Guidelines For Large Fact Tables"](#)

## 3.1 Operating System, Driver, and Connectivity Requirements for Oracle Business Analytics Warehouse

Table 3–1 provides the operating system, driver, and connectivity software requirements for the Oracle Business Analytics Warehouse components.

**Note:** For information about the supported versions of the components listed in Table 3–1, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

**Table 3–1 OS, Driver and Connectivity Requirements for Oracle Business Analytics Warehouse**

Component	Operating System	Software	Connectivity and Driver
ETL Servers	<ul style="list-style-type: none"> <li>■ Solaris</li> <li>■ AIX</li> <li>■ HP</li> <li>■ Windows</li> <li>■ Linux</li> </ul>	<ul style="list-style-type: none"> <li>■ Informatica PowerCenter Integration Service</li> <li>■ DAC Server</li> </ul>	<ul style="list-style-type: none"> <li>■ Java SDK</li> <li>■ JDBC drivers</li> <li>■ Javamail</li> </ul>
ETL Clients	Windows	<ul style="list-style-type: none"> <li>■ Informatica Client tools</li> <li>■ DAC Client</li> </ul>	<ul style="list-style-type: none"> <li>■ Java SDK</li> <li>■ JDBC drivers</li> <li>■ ODBC drivers</li> </ul>
Oracle Business Analytics Warehouse	<ul style="list-style-type: none"> <li>■ Solaris</li> <li>■ AIX</li> <li>■ HP</li> <li>■ Windows</li> <li>■ OS/390, z/OS</li> </ul>	Database software with Oracle Business Analytics Warehouse	Not applicable
ETL Repositories	<ul style="list-style-type: none"> <li>■ Solaris</li> <li>■ AIX</li> <li>■ HP</li> <li>■ Windows</li> </ul>	Database software with ETL repositories	Not applicable

## 3.2 General Guidelines for Setting Up Oracle Business Analytics Warehouse

The Oracle Business Analytics Warehouse is a database that contains dimensional schemas. Although it is technically possible to put the Oracle Business Analytics Warehouse in the same database as the transactional database, it is not recommended for performance reasons. The transactional database is structured as an online transaction processing (OLTP) database, whereas the Oracle Business Analytics Warehouse is structured as an online analytical processing (OLAP) database, each optimized for its own purpose. The reasons for not combining the two databases are:

- The analytical queries interfere with normal use of the transactional database, which is entering and managing individual transactions.
- The data in a transactional database is normalized for update efficiency. Transactional queries join several normalized tables and will be slow (as opposed to pre-joined, de-normalized analytical tables).

- Historical data cannot be purged from a transactional database, even if not required for current transaction processing, because you need it for analysis. (By contrast, the analytical database is the warehouse for historical as well as current data.) This causes the transactional database to further slow down.
- Transactional databases are tuned for one specific application, and it is not productive to use these separate transactional databases for analytical queries that usually span more than one functional application.
- The analytical database can be specifically tuned for the analytical queries and Extract-Transform-Load (ETL) processing. These are quite different from transactional database requirements.
- On the transactional database, you should place the S\_ETL tables in a separate tablespace. These ETL tables are used by the Oracle Business Analytics Warehouse and should not be part of the routine backup processes.

A complete listing of these tables is available in *Oracle Business Analytics Warehouse Data Model Reference*.

- To maximize ETL performance on Siebel OLTP databases, create three indexes on the Siebel OLTP database, using the following SQL commands:

```
CREATE INDEX S_AUDIT_ITEM_M100 ON S_AUDIT_ITEM (FIELD_NAME
ASC, BUSCOMP_NAME DESC) PCTFREE 10 ALLOW REVERSE SCANS
COLLECT DETAILED STATISTICS;
```

```
CREATE INDEX S_AUDIT_ITEM_M101 ON S_AUDIT_ITEM (RECORD_ID ASC,
FIELD_NAME DESC) PCTFREE 10 ALLOW REVERSE SCANS COLLECT
DETAILED STATISTICS;
```

```
CREATE INDEX S_OPTY_M102 ON S_OPTY (ROW_ID ASC, PR_POSTN_ID
DESC) PCTFREE 10 ALLOW REVERSE SCANS COLLECT DETAILED
STATISTICS;
```

The Informatica Repository stores all of the Informatica object definitions for the ETL mappings that populate the Oracle Business Analytics Warehouse. It is a series of repository tables that are stored in a database, which can be transactional, analytical, or a separate database.

The Oracle Business Analytics Warehouse works with relational database management systems. In addition to the general requirements, there are additional database management systems (DBMS)-specific requirements depending on the DBMS you are using.

The following general guidelines will help you set up the data warehouse physical database for performance and growth:

- At a minimum, separate the data and index tablespaces. Create more tablespaces to separate heavily used tables and their indexes.
- Use the maximum block and page size available for tablespaces ((for example, 32K), because it provides good overall performance and also does not impose low limits to the maximum size to which the tablespace can grow, as compared to 4K, 8K,16K sizes.
- If you are using multiple disk storage systems, stripe the tablespace containers and files across as many disks as possible.
- Raw devices for tablespaces provide better performance as compared to cooked file systems.
- RAID-5 is known to give a good balance of performance and availability.

- Size the buffer pools based on content and size (number of tables and their sizes) of tablespaces.
- Allocate about 75 percent of the total available server memory to the database, assuming no other application is running on the same server.

During the Oracle Business Analytics Warehouse configuration process, when you create the data warehouse tables using the procedure [Section 4.11.1, "Creating Data Warehouse Tables,"](#) you can create tables in one tablespace and indexes in another tablespace. However, for performance reasons, it is recommended that you create tablespaces as described in [Table 3–2](#).

**Table 3–2 Recommended Tablespace Configuration**

Tablespace Name	List of Tables
DIM_STG	W_*DS
FACT_STG	W_*FS
DIM	W_*D and W_*MD
FACT	W_*F
AGG	W_*A
OTHER	Remaining W* tables
DIM_INDX	Indexes of W_*D tables (for example, other tables would include W*G and W*GS tables)
FACT_INDX	Indexes of W_*F tables
OTHER_INDX	Remaining indexes of W* tables

---

**Note:** To avoid fatal deadlocks during the ETL, make sure that you select the 'Session Level Retry on Deadlock' option in Informatica.

---

### 3.3 IBM DB2 UDB-Specific Database Guidelines for Oracle Business Analytics Warehouse

[Table 3–3](#) provides guidelines for parameter settings for DB2 relational database management system (RDBMS) usage. Use these guidelines as a starting point. You will need to make changes based on your specific database sizes, data shape, server size (CPU and memory), and type of storage. The database administrator should make changes to the settings based on performance monitoring and tuning considerations.

**Table 3–3 Recommended DB2 Parameter Settings**

Parameter	DB2 UDB V7	DB2 UDB V8	Notes
SHEAPTHRES	400000	400000	
ASLHEAPSZ	15	15	
RQRIOBLK	65535	65535	
QUERY_HEAP_SZ	16384	16384	
JAVA_HEAP_SZ	2048	2048	
MAXAGENTS	400	400	
NUM_INITAGENTS	10	10	

**Table 3–3 (Cont.) Recommended DB2 Parameter Settings**

Parameter	DB2 UDB V7	DB2 UDB V8	Notes
NUM_POOLAGENTS	200	200	
INTRA_PARALLEL	YES	YES	
FCM_NUM_BUFFERS	12288	12288	
SHEAPTHRES_SHR	N/A	=SHEAPTHRES	
DBHEAP	16384	16384	
CATALOGCACHE_SZ	5558	5558	
LOGBUFSZ	2048	2048	
UTIL_HEAP_SZ	10000	10000	
NUM_ESTORE_SEGS	16	NIL	Restore is not needed in DB2 V8 64-bit because the V7 limit of 1.75 GB addressable memory has been lifted.
ESTORE_SEG_SZ	65536	NIL	
LOCKLIST	25000	25000	
APP_CTL_HEAP_SZ	5000	5000	
SORTHEAP	4000	4000	
STMTHEAP	40960	40960	
APPLHEAPSZ	2560	2560	
PCKCACHESZ	2560	2560	
STAT_HEAP_SZ	20000	20000	
DLCHKTIME	10000	10000	
MAXLOCKS	50	50	
LOCKTIMEOUT	1200	1200	
MAXAPPLS	500	500	
AVG_APPLS	10	10	
MAXFILOP	500	500	
GROUPHEAP_RATIO	N/A	70	New in V8
APPGROUP_MEM_SZ	N/A	30000	New in V8
DATABASE_MEMORY	N/A	AUTOMATIC	New in V8

---

**Note:** To avoid fatal deadlocks during the ETL, make sure that you select the 'Session Level Retry on Deadlock' option in Informatica.

---

## 3.4 IBM DB2 UDB zOS and OS/390 and z/OS-Specific Database Guidelines for Oracle Business Analytics Warehouse

The following requirements apply to IBM DB2 RDBMS usage for zOS and OS/390:

- The Oracle BI Applications communicate with IBM DB2 UDB for z/OS and OS/390 (running on zSeries servers) through IBM DB2 Connect middleware.

The following editions of DB2 Connect are supported:

- **DB2 Connect Enterprise Edition (EE).** This edition is installed on a mid-tier server such as an Informatica Server/Client, DAC, and Oracle Business Intelligence.
- **DB2 Connect Unlimited Edition (UE).** This edition provides the functionality of DB2 Connect Enterprise Edition but is priced differently.
- The ODBC driver for all connections must use the IBM DB2 ODBC Driver.
- Make the appropriate connections using the DB2 Client Configuration Assistant.
- Use the variable settings shown in [Table 3–4](#).

**Table 3–4 Variable Settings for IBM DB2 UDB zOS and OS/390 Databases**

Parameter	Recommended Setting	Notes
IDHTOIN	1800	
CDSSRDEF	Any	
STARJOIN	1	This setting indicates that star join is enabled. The one table with the largest cardinality is the fact table. However, if there is more than one table with this cardinality, star join is not enabled.

## 3.5 SQL Server-Specific Database Guidelines for Oracle Business Analytics Warehouse

This section provides guidelines for SQL Server database usage.

---

**Note:** The SQL Server database must be created with a collation sequence that supports binary sort order or case-sensitive dictionary sort order. Case-insensitive dictionary sort order is not supported. For example, for binary sort order with the U.S English character set, use the collation 'Latin1\_General\_BIN'. If you use the default collation setting of 'SQL\_Latin1\_General\_CP1\_CI\_AS', the database is set to case-insensitive, which is not supported, and causes index creation failures.

---

This section includes the following topics:

- [Section 3.5.1, "Setting the ANSI NULL Option"](#)
- [Section 3.5.2, "Modifying the DB Library Options Setting"](#)
- [Section 3.5.3, "Recommended SQL Server Database Parameters"](#)



### 3.5.1 Setting the ANSI NULL Option

Oracle BI Applications requires that SQL Server databases be created with the ANSI NULL option selected.

#### To set the ANSI NULL option

1. In the SQL Server Enterprise Manager, right-click the appropriate database, and choose Properties.
2. Click the Options tab and select the box for ANSI NULL default.

### 3.5.2 Modifying the DB Library Options Setting

In a SQL Server 2000 environment, when loading Analytics tables with international data, or loading more than one language, you need to modify the DB Library Options setting.

#### To modify the DB Library Options setting

1. From the Microsoft SQL Server program menu, select Client Network Utility.
2. Select the DB Library Options tab.
3. Clear the option Automatic ANSI to OEM.

---

**Note:** SQL Server 2000 automatically tunes many of the server configuration options; therefore, an administrator is required to do little, if any, tuning. Although these configuration options can be modified, the general recommendation is that these options be left at their default values, allowing SQL Server to automatically tune itself based on run-time conditions.

---

### 3.5.3 Recommended SQL Server Database Parameters

If necessary, SQL Server components can be configured to optimize performance, as shown in [Table 3-5](#).

**Table 3-5 Recommended Variable Settings for SQL Server Databases**

Parameter	Recommended Setting	Notes
Affinity mask	0	
Allow updates	0	
Awe enabled	0	
C2 audit mode	0	
Cost threshold for parallelism	5	
Cursor threshold	-1	
Default full-text language	1033	
Default language	0	
Fill factor	95%	For insert-intensive transactions, set Fill Factor between 90 and 95%. For better query performance, set Fill factor to 95 or even 100%.
Index create memory	1024 KB	Default is 0.

**Table 3–5 (Cont.) Recommended Variable Settings for SQL Server Databases**

Parameter	Recommended Setting	Notes
Lightweight pooling	0	
Locks	0	
Max degree of parallelism	0	Default is 0. This turns off parallelism. Max degree of parallelism should be left at 0, which means use parallel plan generation. It should be set to 1 (use only 1 process) if you run multi threaded components (for example, several EIM threads).
Max server memory	2000 MB	Default is 2147483647.
Max text repl size	65536 B	
Max worker threads	100	Default is 255.
Media retention	0	
Min memory per query	1024 KB	
Min server memory	500 MB	Default is 0.
Nested triggers	1	
Network packet size	8192 B	Default is 4096.
Open objects	0	
Priority boost	0	
Query governor cost limit	0	Modify to 60 only if CPU is high.
Query wait	-1 sec	
Recovery interval	0 min	
Remote access	1	
Remote login timeout	20 sec	
Remote proc trans	0	
Remote query timeout	600 sec	
Scan for startup procs	0	
Set working set size	0	
Two-digit year cutoff	2049	
User connections	0	
User options	0	

- **SQL Server memory:** Make sure adequate memory is available.
- **Transaction logs and TempDB:** Reside on a separate disk from those used by database data.
- **Full load:** Full Recovery model for the database.
- **Incremental (Refresh) load:** Change from Full to Bulk-Logged Recovery model.

## 3.6 Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse

This sections contains the recommended best practices and guidelines for maximizing performance in Teradata deployments. It contains the following topics:

- [Section 3.6.1, "Installation of Required JDBC Driver for Teradata Databases"](#)
- [Section 3.6.2, "General Guidelines for Teradata Deployments"](#)
- [Section 3.6.3, "Best Practices For Teradata Deployments"](#)

### 3.6.1 Installation of Required JDBC Driver for Teradata Databases

If you have a Unicode environment on a Teradata database, you must install the Teradata JDBC Driver for Teradata 12.0. This driver is required for all supported versions of Teradata databases, including versions earlier than version 12.

The Teradata JDBC Driver for Teradata 12.0 is available at <http://www.teradata.com/DownloadCenter>, under the title Teradata JDBC Driver.

### 3.6.2 General Guidelines for Teradata Deployments

The following requirements apply to Teradata database usage:

- Install the FastLoad and TPump Teradata utilities on the machine on which the Informatica Server is installed.
- When installing Informatica PowerCenter, make sure there are no spaces in the Informatica Server directory name or the directory path. The default directory contains spaces that you must remove manually.
- When you install Oracle BI Applications in a Teradata environment, you must load the prebuilt Informatica Repository file `\dwrep\Oracle_BI_DW_Teradata.rep`.
- In the ODBC configuration for Teradata, set the session Mode to ANSI and DateFormat to AAA, before you create the schema. Tables must be created as case specific. If you do the ODBC configuration after the schema is created, tables might not be created as case specific.
- To avoid overflows if the arithmetic calculation involves any decimal data type more than 18,3 (precision, scale), add the following static Source System Parameters:
  - "as decimal(18,3))" for `$$Hint_Tera_Post_Cast`
  - "Cast (" for `$$Hint_Tera_Pre_Cast` `$$Hint_Tera_Post_Cast` `$$Hint_Tera_Pre_Cast`)

For more information about setting Source System Parameters in DAC, see [Section 4.18.1, "Setting DAC Source System Parameters."](#)

---

**Note:** To avoid fatal deadlocks during the ETL, make sure that you select the 'Session Level Retry on Deadlock' option in Informatica.

---

### 3.6.3 Best Practices For Teradata Deployments

This sections contains the recommended best practises for maximizing performance in Teradata deployments, and contains the following topics:

- [Section 3.6.3.1, "LEFT OUTER JOIN Issue"](#)

- [Section 3.6.3.2.2, "Group By Vs Distinct"](#)
- [Section 3.6.3.2.3, "Pruning Of Mapping"](#)
- [Section 3.6.3.2.4, "Loader Configurations"](#)

---

**Note:** The following best practices should be treated as customizations. You should follow standard customization methodology, such as copying the mapping to a Custom directory. You should never make these changes directly in the default objects.

---

### 3.6.3.1 LEFT OUTER JOIN Issue

Teradata redistributes the OUTER table according to the join key, unless the INNER table is tiny. If the INNER table is tiny, the INNER table could be simply copied to all AMPs and the OUTER table would not be redistributed.

Many fact SIL mappings need to get the dimensional ROW\_WID from ROW\_ID/INTEGRATION\_ID. For example, W\_PER\_RANK\_FS.ACCNT\_ID needs to be converted to ACCNT\_WID before loading into W\_PER\_RANK\_F table. Since ACCT\_ID is nullable, the join between W\_PER\_RANK\_FS and W\_ORG\_D is defined as a LEFT OUTER JOIN.

However, the percentage of NULLs in ACCT\_ID column can be as high as 50% or more depending on the data set. When redistributing the W\_PER\_RANK\_FS according to ACCT\_ID, all rows with ACCT\_ID = NULL are put onto a single AMP.

Although a Teradata database usually has hundreds of gigabytes of spool space, the spool space is allocated across hundreds of AMPs. The spool space for each AMP is limited (for example, to two gigabytes).

When a large percentage of W\_PER\_RANK\_FS is distributed to a single AMP, this can result in insufficient spool space being available. This happens when too much data is spooled onto one AMP, not because the spool space is too small.

To work with Teradata's mechanism of parallel processing, and resolving the LEFT OUTER JOIN, SQL must be re-written.

As an example, refer to the following original SQL:

```
SELECT ... FROM
W_PER_RANK_FS FS LEFT OUTER JOIN W_ORG_D ORG ON
  FS.ACCNT_ID = ORG.INTEGRATION_ID AND
  FS.DATASOURCE_NUM_ID = ORG.DATASOURCE_NUM_ID
```

The above SQL should be re-coded as follows:

```
SELECT ... FROM
W_PER_RANK_FS FS LEFT OUTER JOIN
  (SELECT
    FS.INTEGRATION_ID, FS.DATASOURCE_NUM_ID, ORG.ROW_WID, ORG.GEO_WID
  FROM
    W_PER_RANK_FS FS, W_ORG_D ORG
  WHERE
    FS.ACCNT_ID = ORG.INTEGRATION_ID AND FS.DATASOURCE_NUM_ID =
    ORG.DATASOURCE_NUM_ID AND FS.ACCNT_ID IS NOT NULL)
ORG ON
  FS.DATASOURCE_NUM_ID = ORG.DATASOURCE_NUM_I AND
  FS.INTEGRATION_ID = ORG.INTEGRATION_ID
```

The same SQL re-coding method can be used for other Source Qualifiers that run into spool space problems.

In addition, there is another solution to the lack of spool space issue. This solution is to convert the NULLs to some evenly distributed but non-matched values, as shown in the following SQL example:

```
SELECT ... FROM
W_PER_RANK_FS FS LEFT OUTER JOIN W_ORG_D ORG ON
CASE WHEN FS.ACCNT_ID IS NOT NULL
THEN FS.ACCNT_ID
ELSE '#' || FS.INTEGRATION_ID END
= ORG.INTEGRATION_ID
AND FS.DATASOURCE_NUM_ID = ORG.DATASOURCE_NUM_ID
```

---



---

**Note:** The re-coded SQL is shown in bold type.

---



---

### 3.6.3.2 Additional Statistics Required

Oracle Business Intelligence provides a list of column and index statistics collection statements. These are generally applicable for all situations, but should be evaluated for each site and situation. Additional statistics might be required, depending on your requirements. For example, some workflows need more statistics to avoid an out-of-spool space error on Teradata server.

**3.6.3.2.1 Collect Statistics As A Prerequisite** Once the tables have been created in the staging and target databases, you must run the supplied statistics collection. Failure to do so can affect ETL performance, and possibly result in a spool space error (error number 2646)

DAC does the re-collection of statistics as part of the ETL processes. However, DAC issues the collect statistics statement at the table level only (for example, collect statistics on w\_org\_d), and only for existing statistics.

**3.6.3.2.2 Group By Vs Distinct** When there is a low number of distinct values, it is more efficient to use the GROUP BY phrase. Do not use the DISTINCT phrase, unless the number of distinct values is high.

**3.6.3.2.3 Pruning Of Mapping** If you do not use all of the OTB fields supplied, you can improve performance by stripping the extraneous fields from the mapping and the tables.

**3.6.3.2.4 Loader Configurations** This section explains the loaders that are available in Teradata, and how they are used in Oracle BI Applications.

Teradata has three different types of Teradata loader processes, as follows:

- Fastload
- Mload
- T pump

Each loader process can be used in two different modes, as follows:

- Staged Mode: The Informatica process does the following in this order:
  - Reads from the source data.
  - Creates a data file.
  - Invokes the loader process to load the table using the data file created.

**Advantages:** In the event of failures, you can recover using the Teradata recovery process.

**Disadvantages:** Staged mode is slower than Piped mode, and you need more disk space, as it can create large data files.

- Piped Mode: The Informatica process reads from the source and simultaneously pipes that data to the loader to start loading the target table.

**Advantages:** Quicker than Staged mode, and you do not require large amounts of disk space because no data files are created.

**Disadvantages:** In the event of failures, you cannot recover using the Teradata recovery process (because tpump does row commits unlike fastload and mload).

**3.6.3.2.5 Fastload** You use the Fastload process on empty tables, such as loading staging tables and in initial loads where the tables are empty.

When the Fastload process starts loading, it locks the target table, which means that processes (for example, lookups) cannot access that table. One solution to this problem is to specify dummy SQL for the look up overrides at the session level.

**Tip:** If a session fails during a Fastload process, use SQL Assistant to run a simple SQL command (for example, count(\*)), to determine whether the table is locked by a Fastload process.

If a table is locked (for example, (for W\_ORG\_DS), use the following script to release the lock:

```
LOGON DATABASEALIAS/USER, PASSWORD
BEGIN LOADING USER.W_ORG_DS
ERRORFILES USER.ET_W_ORG_DS, USER.UV_W_ORG_DS;
END LOADING;
```

If you save the above text in a file called test.ctl, you would run this process by entering the following command at a command prompt:

```
C:\fastload\test.ctl
```

**Tip:** To create a load script for a table, edit the test.ctl script above to change the login information, and replace all occurrences of W\_ORG\_DS with the required target table name.

After a load process script runs successfully, you should be able to run the command 'select count(\*)' on the target table. If you are not able release the lock, you might need to drop and re-create the table to remove the lock. If you do so, you must re-create the statistics.

**Tip:** Fastload is typically used in piped mode to load staging tables and initial loads. In the event of errors, reload the entire data.

**3.6.3.2.6 Mload** The Mload process is slower than Fastload but quicker than Tpump. The Mload process can work on both empty tables as well as on tables with data. In the event of errors when running in piped mode, you cannot recover the data.

**3.6.3.2.7 Tpump** The Tpump process is slower than Mload but faster than ODBC. The Tpump process does row commits, which enables you to recover processed operations, even if you use piping mode. In other words, if you re-start the process, Tpump starts loading data from the last committed data.

Tpump can be used in the following modes:

- Tpump\_Insert: Use to do inserts.
- Tpump\_Update: Use to do updates (this mode requires you to define the primary key in the Informatica target table definition).
- Tpump\_Upsert: Use to do update otherwise insert (this mode requires you to define the primary key in the Informatica target table definition).
- Tpump\_Delete: Use to do deletes (this mode requires you to define the primary key in the Informatica target table definition).

Informatica uses the actual target table name to generate the error table and log tables to be used as part of its control file generation. If you have two instances of Tpump loading into same target table at the same time, you need to modify the session to use a different error table and log table name.

The Tpump load process in piped mode is useful for incremental loads, and where the table is not empty. In the event of errors, restart the process and it starts re-loading from the last committed data.

Refer to Informatica documentation for information about configuring a session to use Teradata loaders.

### 3.7 Oracle-Specific Database Guidelines for Oracle Business Analytics Warehouse

To configure the Business Analytics Data Warehouse on Oracle databases more easily, refer to the parameter template file `init10gR2.ora`. This file is located in `<DRIVE>:\<BI Applications install directory>\dwrep\Documentation\`.

The `init.ora` parameter template file provides parameter guidelines based on the cost-based optimizer for Oracle 10g. Use these guidelines as a starting point. You will need to make changes based on your specific database sizes, data shape, server size (CPU and memory), and type of storage. The database administrator should make changes to the settings based on performance monitoring and tuning.

Copy the template file into your `$ORACLE_HOME/dbs` directory. Then, review the recommendations in the template file, and make the changes based on your specific database configuration. The database administrator should make changes to the settings based on performance monitoring and tuning considerations.

### 3.8 Additional Suggestions for Optimizing Oracle Performance in Oracle Business Analytics Warehouse

This section contains additional suggestions for optimizing performance for Oracle databases.

- Oracle BI Applications under Oracle support only binary sorting. If you are running an Oracle client, do one of the following:
  - Set the `NLS_SORT` parameter to `BINARY`.
  - Choose a `NLS_LANG` setting that includes binary.

These settings are required for adequate performance from the dedicated Web client.

- Make sure that cost-based optimization is enabled in the Oracle development, test, and production databases and that statistics are kept up to date. Otherwise, the rule-based optimizer may be used.
- Create foreign keys in the Oracle database, but configure Oracle to not enforce the foreign key relationship. The existence of foreign keys will allow Oracle to better optimize certain queries. By turning off enforcement, the database load should not be negatively affected.
- Analyze application for occurrences of highly skewed data that is indexed. Create histogram statistics for these indexes to enable the optimizer to better perform queries.
- To increase data throughput between Oracle BI Server and Oracle, change SDU and TDU settings in listener.ora. The default is 2 KB and can be increased to 8 KB.
- On the server side, edit the listener.ora file. Under the particular SID\_LIST entry, modify SID\_DESC as follows:

```
SID_LIST_LISTENER =  
  SID_LIST =  
    SID_DESC = (SDU=16384) (TDU=16384)  
              (ORACLE_HOME = /.....)  
              (SID_NAME = SOLAP)  
    )  
  )
```

- Set the tablespace to at least the same as the transactional database size. Make sure the temporary tablespace has adequate space.
- Set the number of log file groups to 4.
- Set the size of each log file to 10 MB.
- The sga\_max\_size to 700 MB.
- On the client side, edit the tnsnames.ora file. Modify the TNS alias by adding SDU= and TDU= as follows:

```
myhost_orcl.world=  
  DESCRIPTION=(SDU=16384) (TDU=16384)  
  ADDRESS = (PROTOCOL = TCP) (HOST=myhost) (PORT=1521)  
  CONNECT_DATA=(SID=ORCL)
```

## 3.9 Miscellaneous Information About Oracle BI Applications Deployments

This section contains the following topics:

- [Section 3.9.1, "Preconfigured Mapping for the SA System Subject Area"](#)
- [Section 3.9.2, "Using Initialization Blocks"](#)
- [Section 3.9.3, "Creating Custom Indexes in Siebel Source Databases for Incremental Load Performance"](#)
- [Section 3.9.4, "Creating Custom Indexes in Oracle EBS Source Database for Incremental Load Performance"](#)
- [Section 3.9.5, "Running an ETL with a Non-English OLTP Data Source"](#)



### 3.9.1 Preconfigured Mapping for the SA System Subject Area

For Oracle's Siebel Applications customers, [Table 3–6](#) describes the preconfigured mappings for the SA system subject area. Fields that are not available in Oracle's Siebel transactional database will default to values shown in the table.

- **Overriding Defaults.** You can add user-specific values for these fields, by creating an extension table to the S\_USER table. to store the user-specific defaults for these fields. Additionally, you can change any of the default values. The metadata for the following logical table can be modified to include any physical extension table.

SA User. (User)

For instructions, refer to the documentation about configuring tables and columns for Oracle's Siebel Business Applications.

- **Setting Provider Information.** Typically, the cell phone and the fax numbers in the Oracle Business Analytics Warehouse do not contain a provider name. Therefore, the Pager will typically be a numeric value such as 555-483-3843. To append a provider to this address, use the following guidelines:
  - If the entire company has the same provider, then you can append the provider in the column mappings.
  - If users can have different providers, you need to create an extension table. For instructions, refer to the documentation about configuring tables and columns for Oracle's Siebel business applications.

**Table 3–6 Preconfigured Mappings for the User Table in the SA System Subject Area**

Logical Column	Physical Table	Expression	Comments
Cell Phone		"	It might be mapped to S_CONTACT.CELL_PH_NUM if this field contains SMTP address.
Cell Phone Priority		"	Defaults to N
Display Name	S_CONTACT	"Real Time OLTP"."SIEBEL.S_CONTACT_User.FST_NAME    ' '    "Real Time OLTP"."SIEBEL.S_CONTACT_User.LAST_NAME	First Name concatenated with Last Name
Email	S_CONTACT	EMAIL_ADDR	
Email Priority		'HNL'	Defaults to N
Email Type		'html'	Defaults to HTML
Group Name	S_RESP	NAME	
Handheld		"	Defaults to an empty string
Handheld Priority		"	Defaults to an empty string
Language		'en'	Defaults to 'en'
Locale		'en'	Defaults to 'en'
Logon	S_USER	LOGIN	

**Table 3–6 (Cont.) Preconfigured Mappings for the User Table in the SA System Subject**

Logical Column	Physical Table	Expression	Comments
Pager		"	It could be mapped to S_CONTACT.PAGER_PH_NUM if this field contains SMTP address
Pager Priority		"	Defaults to N
Time Zone	S_TIMEZONE	NAME	

### 3.9.2 Using Initialization Blocks

Table 3–7 lists some of the initialization blocks common to all Oracle BI Applications and their purposes. Initialization blocks that are specific to each Oracle BI Applications area are not listed here.

**Table 3–7 Initialization Blocks and Their Purposes**

Initialization Block	Purpose
Authorization	Calculates user's responsibilities from the database.
Authentication	Authenticates against the database and verifies the user exists as a operational application user.
External Metadata Strings	Gets the translated value of metadata strings for the user's locale. This initialization block is critical to Intelligence Dashboards in international deployment.
LOCALE	Sets the user's locale specification in the Oracle BI Server.
Login Properties	Gets user's login properties, such as full name, from the database. This initialization block also sets the logging level of the users. By default, the log level is 0 for all users. If you wish to generate Oracle BI query logs for all users, this initialization block should be updated by changing the default value as well as the value in the initialization SQL.
Default System Language ID	Sets the variable OLTP_LANG_ID by querying the operational application database.
Organizations for Org-based Security	Queries the operational application database to get the organization membership for each user. It sets the variable ORGANIZATION.
Primary Owner ID	Sets the primary owner ID based on the user login ID.
Primary Position ID	Queries the operational application database to set the variable PRIMARY_POSTN_ID.
Warehouse Refresh Date	Sets several time-based variables such as CURRENT_YEAR.
ETL Run Date	Retrieves the ETL run date.
ETL Default Currency	Retrieves the default currency.

### 3.9.3 Creating Custom Indexes in Siebel Source Databases for Incremental Load Performance

To maximize performance for Siebel CRM, you can implement indexes using SQL files that are available in the \OracleBI\dwrep directory. Table 3–8 describes the SQL files that are appropriate for the specified applications.

**Table 3–8 SQL files for Siebel transactional databases**

Application Name	SQL File Name
Horizontal Application	PerfIndex_Horizontal.sql
Industry Application	PerfIndex_Industry.sql

The SQL files generate indexes on all S\_.\* tables that are being used by the preconfigured applications.

---

**Note:** If you move from a test to a production environment, you need to drop the indexes and re-create them in the production environment.

---

### 3.9.3.1 An Example of Change Capture SQL and Required Indexes

Change capture SQL generates the following SQL:

```

Insert into S_ETL_I_IMG_XX (ROW_ID, LAST_UPD)
AS
SELECT ROW_ID, LAST_UPD, MODIFICATION_NUM
From
S_XXX
WHERE LAST_UPD > 'LAST_REFRESH_DATE - PRUNE DAYS'
AND NOT EXISTS
(
SELECT 'X' FROM S_ETL_R_IMAGE
WHERE S_ETL_R_IMAGE.ROW_ID = S_XXX.ROW_ID AND
S_ETL_R_IMG_XX.MODIFICATION_NUM = S_XXX.MODIFICATION_NUM AND
S_ETL_R_IMG_XX.LAST_UPD = S_XXX.LAST_UPD
)

```

Table 3–9 shows the indexes that, based on the SQL above, are created on the S\_CONTACT table by the SQL generation scripts.

**Table 3–9 Indexes Created on the S\_CONTACT Table in Siebel transactional database**

Index	Index Column
S_CONTACT_W1	LAST_UPD, ROW_ID, MODIFICATION_NUM
S_CONTACT_W11	LAST_UPD

## 3.9.4 Creating Custom Indexes in Oracle EBS Source Database for Incremental Load Performance

Oracle EBS source database tables contain mandatory LAST\_UPDATE\_DATE columns, which are used by Oracle BI Applications for capturing incremental data changes. Some Oracle EBS source tables used by Oracle BI Applications do not have an index on the LAST\_UPDATE\_DATE column because the index can impede performance of incremental loads.

There are three categories of Oracle EBS tables as they relate to indexes on the LAST\_UPDATE\_DATE column:

- Category 1: Tables that do not have indexes on the LAST\_UPDATE\_DATE column but on which indexes can be created without impeding performance.
- Category 2: Tables that have indexes on LAST\_UPDATE\_DATE columns. These indexes were introduced in Oracle EBS release 12.

- Category 3: Tables that cannot have indexes on the LAST\_UPDATE\_DATE column because performance will be impeded in the Oracle EBS environment.

### 3.9.4.1 Creating Indexes for Category 1 Tables

The following DDL script creates custom indexes on the LAST\_UPDATE\_DATE column for Category 1 tables, that is, tables in all Oracle EBS releases that do not already have this index created and for which there are no known performance implications for creating such indexes.

You should run this DDL script if your source system is Oracle EBS release 11i or release 12

**Note:** If your source system is Oracle EBS release 12, Oracle EBS release 11.5.10, Oracle EBS release 11.5.9 or lower and has been migrated to Oracle Applications Tablespace Model (OATM), then replace <IDX\_TABLESPACE> with APPS\_TS\_TX\_IDX.

The DDL script is as follows:

```
CREATE index AP.OBIEE_AP_INVOICE_PAYMENTS_ALL ON AP.AP_INVOICE_
PAYMENTS_ALL(LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index AP.OBIEE_AP_PAYMENT_SCHEDULES_ALL ON AP.AP_PAYMENT_
SCHEDULES_ALL(LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index AP.OBIEE_AP_INVOICES_ALL ON AP.AP_INVOICES_
ALL(LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index GL.OBIEE_GL_JE_HEADERS ON GL.GL_JE_HEADERS (LAST_
UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index ONT.OBIEE_OE_ORDER_HEADERS_ALL ON ONT.OE_ORDER_
HEADERS_ALL(LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index PER.OBIEE_PAY_INPUT_VALUES_F ON PER.PAY_INPUT_
VALUES_F (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index PER.OBIEE_PAY_ELEMENT_TYPES_F ON PER.PAY_ELEMENT_
TYPES_F (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index PO.OBIEE_RCV_SHIPMENT_LINES ON PO.RCV_SHIPMENT_
LINES (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index PO.OBIEE_RCV_SHIPMENT_HEADERS ON PO.RCV_SHIPMENT_
HEADERS (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index AR.OBIEE_AR_CASH_RECEIPTS_ALL ON AR.AR_CASH_
RECEIPTS_ALL (LAST_UPDATE_DATE)
tablespace <IDX_TABLESPACE>;

CREATE index WSH.OBIEE_WSH_DELIVERY_DETAILS ON WSH.WSH_DELIVERY_
DETAILS (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;

CREATE index WSH.OBIEE_WSH_NEW_DELIVERIES ON WSH.WSH_NEW_
DELIVERIES (LAST_UPDATE_DATE) tablespace <IDX_TABLESPACE>;
```

**Note:**

- Make sure you use FND\_STATS to compute statistics on the newly created indexes and update statistics on newly indexed table columns in the Oracle EBS database.
- All indexes created with the DDL in this section have the prefix OBIEE\_. This prefix does not follow standard Oracle EBS index naming conventions. Therefore,

Autopatch may fail during future upgrades. In such cases, the indexes with the OBIEE\_ prefix should be dropped and Autopatch restarted.

### 3.9.4.2 Creating Indexes for Category 2 Tables

The following DDL creates custom indexes on the LAST\_UPDATE\_DATE column for Category 2 tables, that is, tables for which indexes were introduced for the LAST\_UPDATE\_DATE column in Oracle release 12.

You should run this DDL script if your source system is Oracle EBS release 11i.

**Note:** If your source system is Oracle EBS release 11.5.10, Oracle EBS release 11.5.9 or lower and has been migrated to Oracle Applications Tablespace Model (OATM), then replace <IDX\_TABLESPACE> with APPS\_TS\_TX\_IDX.

The DDL script is as follows:

```
CREATE index PO.RCV_TRANSACTIONS_N23 ON PO.RCV_TRANSACTIONS
(LAST_UPDATE_DATE) INITIAL 4K NEXT 2M MINEXTENTS 1 MAXEXTENTS 50
PCTINCREASE 0 INITRANS 2 MAXTRANS 255 PCTFREE 10 tablespace
<IDX_TABLESPACE>;

CREATE index PO.PO_DISTRIBUTIONS_N13 ON PO.PO_DISTRIBUTIONS_ALL
(LAST_UPDATE_DATE) INITIAL 4K NEXT 2M MINEXTENTS 1 MAXEXTENTS 50
PCTINCREASE 0 INITRANS 2 MAXTRANS 255 PCTFREE 10 tablespace
<IDX_TABLESPACE>;

CREATE index PO.PO_LINE_LOCATIONS_N11 ON PO.PO_LINE_LOCATIONS_
ALL (LAST_UPDATE_DATE) INITIAL 4K NEXT 2M MINEXTENTS 1
MAXEXTENTS 50 PCTINCREASE 0 INITRANS 2 MAXTRANS 255 PCTFREE 10
tablespace <IDX_TABLESPACE>;

CREATE index PO.PO_LINES_N10 ON PO.PO_LINES_ALL (LAST_UPDATE_
DATE) INITIAL 4K NEXT 4K MINEXTENTS 1 MAXEXTENTS 50 PCTINCREASE
0 INITRANS 2 MAXTRANS 255 PCTFREE 10 tablespace <IDX_
TABLESPACE>;

CREATE index PO.PO_REQ_DISTRIBUTIONS_N6 ON PO.PO_REQ_
DISTRIBUTIONS_ALL (LAST_UPDATE_DATE) INITIAL 4K NEXT 250K
MINEXTENTS 1 MAXEXTENTS 50 PCTINCREASE 0 INITRANS 4 MAXTRANS 255
PCTFREE 10 tablespace <IDX_TABLESPACE>;

CREATE index PO.PO_REQUISITION_LINES_N17 ON PO.PO_REQUISITION_
LINES_ALL (LAST_UPDATE_DATE) INITIAL 4K NEXT 250K MINEXTENTS 1
MAXEXTENTS 50 PCTINCREASE 0 INITRANS 4 MAXTRANS 255 PCTFREE 10
tablespace <IDX_TABLESPACE>;

CREATE index PO.PO_HEADERS_N9 ON PO.PO_HEADERS_ALL (LAST_UPDATE_
DATE) INITIAL 4K NEXT 1M MINEXTENTS 1 MAXEXTENTS 50 PCTINCREASE
0 INITRANS 2 MAXTRANS 255 PCTFREE 10 tablespace <IDX_
TABLESPACE>;

CREATE index PO.PO_REQUISITION_HEADERS_N6 ON PO.PO_REQUISITION_
HEADERS_ALL (LAST_UPDATE_DATE) INITIAL 4K NEXT 250K MINEXTENTS 1
MAXEXTENTS 50 PCTINCREASE 0 INITRANS 4 MAXTRANS 255 PCTFREE 10
tablespace <IDX_TABLESPACE>;

CREATE index AR.RA_CUSTOMER_TRX_N14 ON AR.RA_CUSTOMER_TRX_ALL
(LAST_UPDATE_DATE) INITIAL 4K NEXT 4M MINEXTENTS 1 MAXEXTENTS 50
PCTINCREASE 0 INITRANS 4 MAXTRANS 255 PCTFREE 10 tablespace
<IDX_TABLESPACE>;
```

**Note:** Make sure you use FND\_STATS to compute statistics on the newly created indexes and update statistics on newly indexed table columns in the Oracle EBS database.

### 3.9.5 Running an ETL with a Non-English OLTP Data Source

To run an ETL with a non-English OLTP data source, you must make a copy of an appropriate source system container, and configure the language, country, and continent parameters.

#### To run an ETL with a non-English OLTP data source

1. In DAC, select File, then New Source System to display the New Source System Container dialog box.
2. Select the Create as a Copy of Existing Container radio button.
3. From the Existing Containers drop-down list, select the container that you want to copy, then click OK.
4. Go to the Design view.
5. Ensure that you have selected the correct container from the Containers drop-down list.
6. Select the Source System Parameters tab.
7. Use the Edit tab below the list of Source System Parameters to change the value of the following parameters in the list:
  - \$\$DFLT\_LANG (Default Language) – For example, for a Japanese data source, change this value to JPN.
  - (optional) \$\$DFLT\_COUNTRY (Default Country)
  - (optional) \$\$DFLT\_CONTINENT (Default Continent)

**Tip:** To find out the value to specify for the \$\$DFLT\_LANG parameter, issue the query 'select VAL from S\_SYS\_PREF where SYS\_PREF\_CD=<ETL value>' against the OLTP database. For example, to find out the default ETL language, issue the following command:

```
select VAL from S_SYS_PREF where SYS_PREF_CD='ETL Default Language' ;
```

8. Create a new ETL plan for the new source system container, and edit its parameters, as follows:
  - a. Click the Execute tab.
  - b. Click the Execution Plans subtab.
  - c. Click New to create a new blank execution tab and use the subtabs below (for example, Subject Areas, Parameters, Ordered Tasks) to specify the execution plan details.
  - d. Click Save.
9. Click Run Now to run the new ETL plan (or use the Schedule tab to specify when you want to run the new execution plan).

## 3.10 Partitioning Guidelines For Large Fact Tables

This section explains how to use partitioning to maximize performance in your Oracle BI Applications deployment, and contains the following topics:

- [Section 3.10.1, "Introduction to Partitioning Large Fact Tables"](#)
- [Section 3.10.2, "How to Partition Large Fact Tables"](#)
- [Section 3.10.3, "Configure DAC to Support ETL for Partitioned Tables"](#)
- [Section 3.10.4, "Example XML Files"](#)

### 3.10.1 Introduction to Partitioning Large Fact Tables

Taking advantage of range partitioning for fact tables will not only reduce index and statistics maintenance time during ETL, but also improve web queries performance. Since the majority of inserts and updates impact the last partition(s), you would need to disable only local indexes on a few impacted partitions, and then rebuild disabled indexes after the load and compute statistics on updated partitions only. Online reports and dashboards should also render results faster, since the optimizer would build more efficient execution plans using partitions elimination logic.

Large fact tables, with more than 10 million rows, are good candidates for partitioning. To build an optimal partitioned table with reasonable data distribution, you can consider partitioning by month, quarter, year, and so on. You can either identify and partition target fact tables before initial run, or convert the populated tables into partitioned objects after the full load.

To implement the support for partitioned tables in Oracle Business Analytics Data Warehouse, you need to update DAC metadata and manually convert the candidates into partitioned tables in the target database.

#### To deploy partitioned fact tables

1. Partition the Large Fact Tables. For more information, see [Section 3.10.2, "How to Partition Large Fact Tables."](#)
2. Configure DAC to support ETL for partitioned tables. For more information, see [Section 3.10.3, "Configure DAC to Support ETL for Partitioned Tables."](#)

### 3.10.2 How to Partition Large Fact Tables

If you have large fact tables that are affecting performance, you can maximize performance by partitioning the fact tables as described in this section.

1. Identify a partitioning key and decide on a partitioning interval.

Choosing the correct partitioning key is the most important factor for effective partitioning, since it defines how many partitions will be involved in web queries or ETL updates. Review the following guidelines for selecting a column for a partitioning key:

- Identify eligible columns of type DATE for implementing range partitioning.
- Connect to the Oracle BI Server repository and check the usage or dependencies on each column in the logical and presentation layers.
- Analyze the summarized data distribution in the target table by each potential partitioning key candidate and data volumes per time range, month, quarter or year.

- Based on the compiled data, decide on the appropriate partitioning key and partitioning range for your future partitioned table.
- The recommended partitioning range for most implementations is a month.

The following table summarizes the recommended partitioning keys for some large Oracle BI Applications fact tables:

**Table 3–10 Recommended Partitioning Keys for Large Fact Tables**

Area	Table Name	Partitioning Key
Financials	W_AP_XACT_F	POSTED_ON_DT_WID
Financials	W_AR_XACT_F	POSTED_ON_DT_WID
Financials	W_GL_REVN_F	POSTED_ON_DT_WID
Financials	W_GL_COGS_F	POSTED_ON_DT_WID
Financials	W_TAX_XACT_F	ACCT_PERIOD_END_DT_WID
Financials	W_GL_OTHER_F	ACCT_PERIOD_END_DT_WID
Sales	W_SALES_INVOICE_LINE_F	INVOICED_ON_DT_WID
Sales	W_SALES_SCHEDULE_LINE_F	ORDERED_ON_DT_WID
Sales	W_REVN_F	CLOSE_DT_WID

**2. Identify future global and local indexes.**

Determine all active indexes on a partitioned table candidate (W\_REVN\_F in the example) using the following query in the DAC schema:

```
select I.NAME, C.NAME, C.POSTN, I.ISUNIQUE, I.ISBITMAP
from W_ETL_TABLE T, W_ETL_INDEX I, W_ETL_INDEX_COL C
where T.ROW_WID = I.TABLE_WID
and T.NAME='W_REVN_F'
and I.ROW_WID=C.INDEX_WID
and I.INACTIVE_FLG = 'N'
order by I.NAME, C.POSTN;
```

Review the query output and choose an existing unique index as a global unique index for your partitioned table. All bitmap indexes must be re-created as local partitioned indexes. If the query returns B-Tree indexes other than the chosen candidate for a global index, consider creating them as local indexes as well.

**3. Create a partitioned table.**

You can pre-create a partitioned table prior to the initial load, or load data into the regular table and then create its partitioned copy and migrate the summarized data. If you have already completed the initial load into a regular table and then decided to partition it, you do **not** need to re-run the initial load.

You can consider two options to convert a table into a partitioned one:

- Create table as select.



This method is simpler and faster.

- Create table exchange partition syntax and then split partitions.

This method is suitable in high availability data warehouses when you have to carry out partitioning with end users accessing the data.

---

**Note:** It is important to make sure that your partition names include the values from the DAC system variables `CURRENT_MONTH_WID` and `PREVIOUS_MONTH_WID`, which are defined in the next section. These variables will be used in custom XML files to handle local and global indexes using syntax: 'partition PART\_@DAC\_CURRENT\_MONTH\_WID' and 'partition PART\_@DAC\_PREVIOUS\_MONTH\_WID'. Note that the DDL syntax must have the prefix 'DAC\_' in the custom XML files.

---

You should create enough partitions to cover the time range for the summarized data. You should create additional partitions for future ranges as well. You may also consider distributing the partitions across different tablespaces.

The syntax in the following SQL commands assumes everything in one tablespace `USERS`:

```
SQL> rename W_REVN_F to W_REVN_F_ORIG;
SQL> create table W_REVN_F
partition by range (CLOSE_DT_WID) (
partition PART_MIN    values less than (200601),
partition PART_200601 values less than (200602),
partition PART_200602 values less than (200603),
...
partition PART_200711 values less than (200712),
partition PART_200712 values less than (200801),
...
partition PART_MAX    values less than (maxvalue)
)
tablespace USERS
nologging parallel
enable row movement
as
select * from W_REVN_F_ORIG
;
```

4. Drop all indexes on the renamed table.

To ensure index naming standards in the target schema, drop all indexes on the renamed table (`W_REVN_F_ORIG` table in the example).

5. Create a global unique index.

Create a global unique B-Tree index using the information from item Step 2, for example:

```
create unique index W_REVN_F_P1
on W_REVN_F (ROW_WID)
tablespace USERS_IDX global nologging parallel;
```

6. Create local indexes.

You can generate the set of required 'Create index' DDL commands for Bitmap indexes using the DAC Repository, for example:

```

select 'CREATE BITMAP INDEX ' || i.name || ' ON W_REVN_F (' || c.name || ' ASC)
tablespace USERS_IDX LOCAL NOLOGGING PARALLEL;'
from W_ETL_TABLE T, W_ETL_INDEX I, W_ETL_INDEX_COL C
where T.ROW_WID = I.TABLE_WID
and T.NAME='W_REVN_F'
and I.ROW_WID = C.INDEX_WID
and I.ISBITMAP = 'Y'
and I.INACTIVE_FLG = 'N';

```

If there are any local B-Tree indexes identified in Step 2, create them as local partitioned indexes as well. You can modify the query above and add `I.ISBITMAP = 'N'`. Connect to the Target schema and execute the generated DDLs to create the indexes.

#### 7. Compute statistics.

Compute statistics on the newly partitioned table and its indexes using the `DBMS_STATS` API, for example:

```

SQL> BEGIN
DBMS_STATS.GATHER_TABLE_STATS( NULL
, tabname=>'W_REVN_F'
, cascade => TRUE
, estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE
, method_opt => 'FOR ALL INDEXED COLUMNS SIZE AUTO'
);
END;

```

#### 8. Enable row movement.

Since the custom XML scripts will make local indexes unusable during incremental runs, you need to set the parameter `skip_unusable_indexes` to true in the session running ETL. Connect to your Informatica Workflow Manager, navigate to Connections, then Relational, and edit Data Warehouse connection. Add the following SQL statement in the field Environment SQL:

```
alter session set skip_unusable_indexes=true;
```

#### 9. Drop the original table.

After you confirm performance gains with the partitioned table and indexes in your target database, you can consider dropping the original renamed table (`_ORIG`).

You now need to configure DAC to support ETL for partitioned tables. For more information, see [Section 3.10.3, "Configure DAC to Support ETL for Partitioned Tables."](#)

### 3.10.3 Configure DAC to Support ETL for Partitioned Tables

Once you have partitioned your fact tables as described in [Section 3.10.2, "How to Partition Large Fact Tables,"](#) you need to configure DAC to support the partitioned tables.

#### To configure DAC to support partitioned tables

##### 1. Inactivate all indexes on a partitioned target table, as follows:

- a. In DAC, go to the Design View.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

- b. Click the Indices tab.

- c. Click the Query button, then enter the table name, then click Go.
- d. Select the Inactive check box for all indexes on the target table.

These indexes will be handled outside DAC through pre-SQL and post-SQL at a task level.

---

**Note:** The DAC version released with Oracle BI Applications release 7.9.5.1 does not disable computing statistics at a table level. Since computing statistics for a partitioned table will be handled by a post-SQL XML file, Analyze Table Task in DAC for the table will be redundant. To workaroud this limitation, you can abort the execution plan in DAC, mark the task Analyze Table for your staging table as Completed and restart the Execution Plan.

---

2. Create two new source system parameters in DAC, as follows:

- a. In DAC, go to the Design View.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

- b. Select your custom container from the pull-down menu (next to the Execute Button).

---

**Note:** You cannot create any custom parameters in the pre-packaged containers. You must first create a new container. Refer to the *Oracle Business Intelligence Applications Database Administration Console Guide* for information about how to create a Custom Container as a copy of a packaged one.

---

- c. Click the Source System Parameters tab in the upper right pane.
- d. Use the New button to create two new source system parameters with the following values:

**Table 3–11 Source System Parameters for Partitioned Tables**

Name	Data Type	Value
CURRENT_MONTH_WID	SQL	select to_char(row_wid) from w_month_d where w_current_cal_month_code = 'Current'  When you click on the Value field to display the Enter Parameter Value dialog box, select DBConnection_OLAP from the <b>Logical Data Source</b> drop down list.
PREVIOUS_MONTH_WID	SQL	select to_char(row_wid) from w_month_d where w_current_cal_month_code = 'Previous'  When you click on the Value field to display the Enter Parameter Value dialog box, select DBConnection_OLAP from the <b>Logical Data Source</b> drop down list.

These source system parameters will be used in pre-SQL and post-SQL XML files during incremental loads.

3. Prepare pre-SQL XML files and post-SQL XML files for full and incremental loads.

You can either obtain the template file `samplesql.xml`, located under `<DAC_HOME>\CustomSQLs` directory and generate the required pre- and post-SQL XML files, or use the XML files examples in the next section for your custom XML files. DAC will execute the SQL from pre-SQL XMLs before Informatica tasks, and then run the SQL from post-SQL XML files.

Create the following set of custom XML files, one set per partitioned table:

- `pre_<TABLE_NAME>_ini.xml` - drops all local and global indexes.
- `post_<TABLE_NAME>_ini.xml` - creates all local and global indexes.
- `pre_<TABLE_NAME>incr.xml` - makes local indexes unusable for last two partitions 'PART@DAC\_CURRENT\_MONTH\_WID' and 'PART\_@DAC\_PREVIOUS\_MONTH\_WID'.
- `post_<TABLE_NAME>incr.xml` - rebuilds the unusable local indexes and computes statistics on the two partitions, the rebuilt local and global indexes.

Important: Make sure that your partition names include the values from the DAC system variables `CURRENT_MONTH_WID` and `PREVIOUS_MONTH_WID`. These variables must be used in the custom XML files to handle local and global indexes using syntax: 'partition PART\_@DAC\_CURRENT\_MONTH\_WID' and 'partition PART\_@DAC\_PREVIOUS\_MONTH\_WID'. Note that the DDL syntax must have the prefix 'DAC\_' in the custom XML files.

4. Copy all pre- and post-SQL XML files under `<DAC_HOME>\CustomSQLs` location.
5. Associate the four generated custom XML files with a task, which loads data into the target partitioned table.
  - a. In DAC, go to the Design View.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)
  - b. Select your custom container from the pull-down menu (next to the Execute Button).
  - c. Click the Tables menu in the upper right pane, and find or query the target partitioned fact table.
  - d. Click the menu Target for Tasks (RO) in the lower right pane.
  - e. Find the SIL primary mapping name. For example, the primary mapping for `W_GL_REVN_F` fact table is `SIL_GLRevenueFact`. Copy or save the mapping name.
  - f. Click the menu Tasks in the upper right pane.
  - g. Find or query the identified mapping name from the preceding step.
  - h. Fill in the custom XML file names in the following fields on the bottom right pane and save the change:

**Table 3–12 Field Values on Tasks Tab**

Field Name on Tasks Tab	Value	Description
Pre-SQL for Full Load	pre_<TABLE_NAME>_ini.xml	Pre SQL xml file name for Full Load. For an example XML file, see <a href="#">Section 3.10.4.1, "Pre_W_REVN_F_ini.xml - Pre SQL XML File Name for Full Load"</a> .
Post-SQL for Full Load	post_<TABLE_NAME>_ini.xml	Post SQL xml file name for Full Load. For an example XML file, see <a href="#">Section 3.10.4.2, "Post_W_REVN_F_ini.xml - Post SQL XML File Name for Full Load"</a> .
Pre-SQL for Incremental Load	pre_<TABLE_NAME>incr.xml	Pre SQL xml file name for Incremental Load. For an example XML file, see <a href="#">Section 3.10.4.3, "Pre_W_REVN_F_incr.xml- Pre SQL XML File Name for Incremental Load"</a> .
Post-SQL for Incremental Load	post_<TABLE_NAME>incr.xml	Post SQL xml file name for Incremental Load. For an example XML file, see <a href="#">Section 3.10.4.4, "Post_W_REVN_F_incr.xml- Post SQL XML File Name for Incremental Load"</a> .

6. Save the changes in DAC.
7. Select the check box Analyze Truncate Table Only under the Execution Plan tab in DAC.
  - a. Go to the Execute view.
  - b. Click the tab Execution Plans in the upper right pane.
  - c. Find your Execution Plan.
  - d. Check the checkbox Analyze Truncated Tables only and save the changes.  
DAC will compute statistics on tables that were truncated prior to an ETL and reloaded during the plan execution.
  - e. Rebuild the Execution Plan in DAC.  
Click the tab Build in the upper right pane to rebuild the execution plan. When the Build process is complete, the ETL is ready to run.

### 3.10.4 Example XML Files

This section contains example XML files for pre-SQL and post-SQL for use in Step 5 of the task [Section 3.10.3, "Configure DAC to Support ETL for Partitioned Tables"](#), and contains the following topics:

- [Section 3.10.4.1, "Pre\\_W\\_REVN\\_F\\_ini.xml - Pre SQL XML File Name for Full Load"](#)
- [Section 3.10.4.2, "Post\\_W\\_REVN\\_F\\_ini.xml - Post SQL XML File Name for Full Load"](#)
- [Section 3.10.4.3, "Pre\\_W\\_REVN\\_F\\_incr.xml- Pre SQL XML File Name for Incremental Load"](#)
- [Section 3.10.4.4, "Post\\_W\\_REVN\\_F\\_incr.xml- Post SQL XML File Name for Incremental Load"](#)

**3.10.4.1 Pre\_W\_REVN\_F\_ini.xml - Pre SQL XML File Name for Full Load**

```

<?xml version="1.0" ?>
<CustomSQLs>
  <!-- Start Drop Index -->
  <!-- Drop W_REVN_F_F1 -->
  <sql name="Drop Index W_REVN_F_F1" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
    <![CDATA[ drop index W_REVN_F_F1 ]]>
  </sql>
  <!-- Drop W_REVN_F_F2 -->
  <sql name="Drop Index W_REVN_F_F2" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
    <![CDATA[ drop index W_REVN_F_F2 ]]>
  </sql>
  ...
  ...
  ...
  <!-- Drop W_REVN_F_U1 -->
  <sql name="Drop Index W_REVN_F_U1" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
    <![CDATA[ drop index W_REVN_F_U1 ]]>
  </sql>
</CustomSQLs>

```

**3.10.4.2 Post\_W\_REVN\_F\_ini.xml - Post SQL XML File Name for Full Load**

```

<?xml version="1.0" ?>
<CustomSQLs>
  <!-- Start Create Index -->
  <!-- Create global index W_REVN_F_U1 -->
  <sql name="Create Index W_REVN_F_P1" type="SQL" continueOnFail="false"
validDBPlatforms="Oracle">
    <![CDATA[ create unique index W_REVN_F_P1 on W_REVN_F(ROW_WID) tablespace
USERS_IDX GLOBAL NOLOGGING PARALLEL ]]>
  </sql>
  <!-- Create local index W_REVN_F_F1 -->
  <sql name="Create Index W_REVN_F_F1" type="SQL" continueOnFail="false"
validDBPlatforms="Oracle">
    <![CDATA[ CREATE BITMAP INDEX W_REVN_F_F1 ON W_REVN_F(COLUMN_NAME1 ASC)
TABLESPACE USERS_IDX LOCAL NOLOGGING PARALLEL ]]>
  </sql>
  <!-- Create local index W_REVN_F_F2 -->
  <sql name="Create Index W_REVN_F_F2" type="SQL" continueOnFail="false"
validDBPlatforms="Oracle">
    <![CDATA[ CREATE BITMAP INDEX W_REVN_F_F2 ON W_REVN_F(COLUMN_NAME2 ASC)
TABLESPACE USERS_IDX LOCAL NOLOGGING PARALLEL ]]>
  </sql>
  ...
  ...
  ...
</CustomSQLs>

```

**3.10.4.3 Pre\_W\_REVN\_F\_incr.xml- Pre SQL XML File Name for Incremental Load**

```

<?xml version="1.0" ?>
<CustomSQLs>
  <!-- Start W_REVN_F_F1 -->
  <sql name="W_REVN_F_F1 Disable Previous" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
    <![CDATA[ alter index W_REVN_F_F1 modify partition PART_@DAC_PREVIOUS_MONTH_

```

```

WID unusable ]]>
  </sql>
  <sql name="W_REVN_F_F1 Disable Current" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
  <![CDATA[ alter index W_REVN_F_F1 modify partition PART_@DAC_CURRENT_MONTH_
WID unusable ]]>
  </sql>
  <!-- Start W_REVN_F_F2 -->
  <sql name="W_REVN_F_F2 Disable Previous" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
  <![CDATA[ alter index W_REVN_F_F2 modify partition PART_@DAC_PREVIOUS_MONTH_
WID unusable ]]>
  </sql>
  <sql name="W_REVN_F_F2 Disable Current" type="SQL" continueOnFail="true"
validDBPlatforms="Oracle">
  <![CDATA[ alter index W_REVN_F_F2 modify partition PART_@DAC_CURRENT_MONTH_
WID unusable ]]>
  </sql>
  ...
  ...
  ...
</CustomSQLs>

```

#### 3.10.4.4 Post\_W\_REVN\_F\_incr.xml- Post SQL XML File Name for Incremental Load

```

<?xml version="1.0" ?>
<CustomSQLs>
  <!-- Start W_REVN_F PART_PREVIOUS_MONTH_WID -->
  <sql name="W_REVN_F Rebuild Previous" type="SQL" continueOnFail="false"
validDBPlatforms="Oracle">
  <![CDATA[ alter table W_REVN_F modify partition PART_@DAC_PREVIOUS_MONTH_WID
rebuild unusable local indexes]]>
  </sql>
  <!-- Start W_REVN_F PART_CURRENT_MONTH_WID -->
  <sql name=" W_REVN_F Rebuild Current " type="SQL" continueOnFail="false"
validDBPlatforms="Oracle">
  <![CDATA[alter table W_REVN_F modify partition PART_@DAC_CURRENT_MONTH_WID
rebuild unusable local indexes]]>
  </sql>
  <!-- Gather Stats on global indexes -->
  <sql name="Gather Stats on W_REVN_F_U1" type="Stored Procedure"
continueOnFail="false" validDBPlatforms="Oracle">
  <![CDATA[
      DBMS_STATS.GATHER_INDEX_STATS( NULL
      , indname => 'W_REVN_F_P1
      , estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE
      , degree => DBMS_STATS.DEFAULT_DEGREE
      )
  ]]>
  </sql>
  <!-- Gather Stats on W_REVN_F table -->
  <sql name="W_REVN_F Gather Stats Previous" type="Stored Procedure"
continueOnFail="false" validDBPlatforms="Oracle">
  <![CDATA[
      DBMS_STATS.GATHER_TABLE_STATS( NULL
      , tabname=>'W_REVN_F'
      , cascade => TRUE
      , partname => 'PART_@DAC_PREVIOUS_MONTH_WID'
      , estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE
      , granularity => 'PARTITION'
  ]]>

```

```
        , method_opt => 'FOR ALL INDEXED COLUMNS SIZE AUTO'
        , degree => DBMS_STATS.DEFAULT_DEGREE
    )
  ]]>
</sql>
<sql name="W_REVN_F Gather Stats Current" type="Stored Procedure"
continueOnFail="false" validDBPlatforms="Oracle">
  <![CDATA[
    DBMS_STATS.GATHER_TABLE_STATS( NULL
    , tabname=>'W_REVN_F'
    , cascade => TRUE
    , partname => 'PART_@DAC_CURRENT_MONTH_WID'
    , estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE
    , granularity => 'PARTITION'
    , method_opt => 'FOR ALL INDEXED COLUMNS SIZE AUTO'
    , degree => DBMS_STATS.DEFAULT_DEGREE
    )

  ]]>
</sql>
</CustomSQLs>
```



---

---

# Installing and Setting Up Oracle BI Applications

This section explains how to install and set up the Oracle BI Applications components. It contains the following main topics:

- Section 4.1, "About Oracle BI Applications Topologies"
- Section 4.2, "High-Level Installation and Configuration Task List"
- Section 4.3, "Mandatory Requirements and Preinstallation Tasks"
- Section 4.4, "Running the Oracle BI Applications Installer on Windows"
- Section 4.5, "Installing Informatica PowerCenter Services Software"
- Section 4.6, "Restoring the Prebuilt Informatica Repository"
- Section 4.7, "Configuring Informatica PowerCenter Components"
- Section 4.8, "Installing an Additional Instance of the DAC Client on Windows"
- Section 4.9, "Configuring the DAC Client"
- Section 4.10, "Logging into DAC for the First Time and Creating the DAC Repository"
- Section 4.11, "Creating the Oracle Business Analytics Warehouse Tables"
- Section 4.12, "Installing the DAC Server"
- Section 4.13, "Configuring the DAC Server"
- Section 4.14, "Configuring DAC Integration Settings"
- Section 4.15, "Configuring Relational Connections in Informatica Workflow Manager"
- Section 4.16, "Configuring the SiebelUnicodeDB Custom Property"
- Section 4.17, "Setting Up DAC to Receive Email Notification"
- Section 4.18, "Additional Configuration Tasks"
- Section 4.19, "About Running A Full Load ETL"

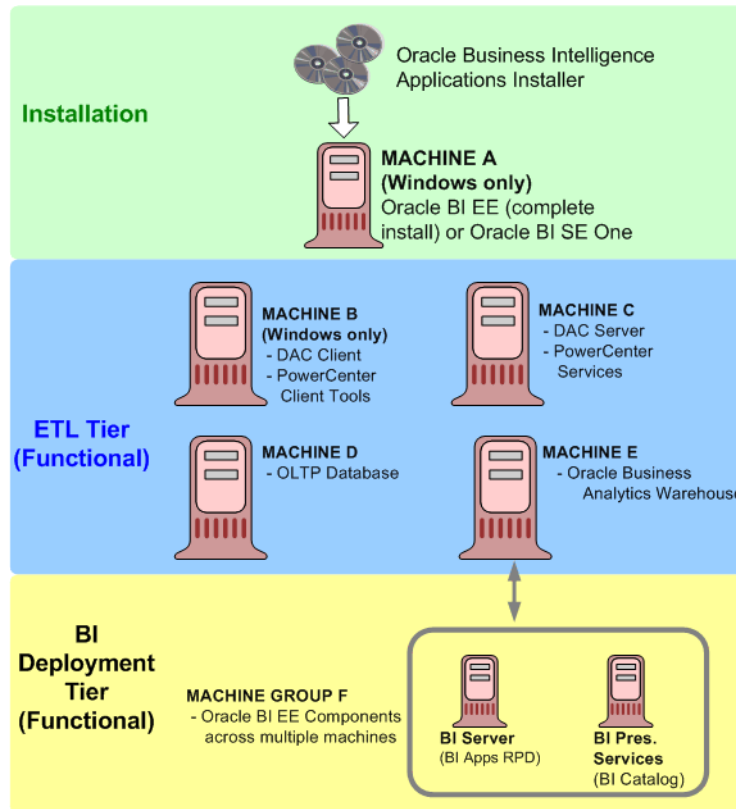
For information about supported Windows versions, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

## 4.1 About Oracle BI Applications Topologies

Oracle BI Applications and Informatica PowerCenter can be deployed flexibly across a wide range of topologies on different platforms and combinations of platforms.

This section uses an example to explain a typical topology for an Oracle BI Applications deployment. Figure 4-1 shows a typical deployment topology.

**Figure 4-1 Typical Topology for an Oracle BI Applications Deployment**



In Figure 4-1, note the following:

- **Installation**

- Machine A (Windows-only)

Machine A is a machine that has installed Oracle Business Intelligence Enterprise Edition, on which you run the Oracle BI Applications installer to install the Oracle BI Applications files.

**Note:** The instance of Oracle Business Intelligence Enterprise Edition does not need to be the functional version of Oracle Business Intelligence Enterprise Edition that you will use to deploy dashboards in your live system. This instance is only required to enable the Oracle BI Applications installer to install the Oracle BI Applications files on a machine.

After the Oracle BI Applications files have been installed on Machine A, the DAC Client is deployed to Machine B, and the DAC Server is deployed to Machine C.

In addition, the following files are copied from the installation machine (Machine A) to the Business Intelligence Deployment Tier (Machine Group F) as follows:

- The OracleBI\Server\Repository\OracleBIAnalyticsApps.rpd file is copied from Machine A to the machine that runs the BI Server in Machine Group F.
- The OracleBIData\Web\Catalog\EnterpriseBusinessAnalytics\\*. \* files are copied from Machine A to the machine that runs the BI Presentation Services Catalog in Machine Group F.

- **ETL Tier (Functional)**

- Machine B (Windows-only)  
Runs the DAC Client and Informatica PowerCenter Client Tools.
- Machine C (Windows, UNIX, Linux)  
Runs the DAC Server and Informatica PowerCenter Services.
- Machine D (Windows, UNIX, Linux)  
Hosts the transactional (OLTP) database.
- Machine E (Windows, UNIX, Linux)  
Hosts the Oracle Business Analytics Warehouse database.

- **BI Deployment Tier (Functional)**

The BI Deployment tier is used to deploy the business intelligence dashboards.

- Machine Group F (Windows, UNIX, Linux)  
Machine Group F is a group of machines that runs the Oracle Business Intelligence Enterprise Edition components. For example, one machine might run the BI Server and BI Applications Repository, and another machine might run the BI Presentation Services.

## 4.2 High-Level Installation and Configuration Task List

This section contains a high-level task list of the tasks you must complete to install Oracle BI Applications. You must complete these tasks in the order listed below.

1. Before you start to install Oracle BI Applications and Informatica PowerCenter Services, do the following:

- a. Make sure that a complete installation of Oracle Business Intelligence Enterprise Edition is installed on the machine on which you want to install Oracle BI Applications. For more information, see [Section 4.3.1, "Oracle Business Intelligence Infrastructure Requirements."](#)

**Note:** This instance of Oracle Business Intelligence Enterprise Edition does not need to be the functional version that you will use to deploy dashboards in your live system. This instance is only required to enable the Oracle BI Applications installer to install the Oracle BI Applications files on a machine.

- b. Make sure that you satisfy the Informatica PowerCenter requirements that are specified in [Section 4.3.2, "Informatica PowerCenter Requirements."](#)

Also, read the Informatica PowerCenter overview section and plan your Informatica topology before you install Informatica PowerCenter.

- c. Make sure that you satisfy the Code Page requirements that are specified in [Section 4.3.3, "Code Page and Data Movement Requirements."](#)

- d. Make sure that you perform the mandatory preinstallation tasks that are specified in [Section 4.3.4, "Preinstallation Tasks."](#) The preinstallation tasks are the following:

[Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"](#)

Create the required databases to store the Informatica Repository, DAC Repository, and Oracle Business Analytics Warehouse.

[Section 4.3.4.2, "Install Database Connectivity Software"](#)

Native connectivity client software is required on machines that will host the Informatica PowerCenter Services, the DAC Client, and the DAC Server. The machines that host these components require connectivity to the Oracle Business Analytics Warehouse (target) database, the transactional (source) databases, and the databases for the DAC Repository and Informatica Repository.

[Section 4.3.4.3, "Create ODBC Database Connections"](#)

The DAC Client uses an ODBC connection to create and alter tables in the Oracle Business Analytics Warehouse. Create an ODBC connection to the Oracle Business Analytics Warehouse database on the machine that hosts the DAC Client.

[Section 4.3.4.4, "Set Environment Variables"](#)

Before you run the Informatica PowerCenter installer, you must set environment variables.

[Section 4.3.4.5, "Download and Install the Java JDK"](#)

Java JDK is required on the machines that host the DAC Client and DAC Server.

[Section 4.3.4.6, "Download Hibernate Libraries"](#)

The DAC Client and DAC Server require libraries from Hibernate, an open source software. Hibernate libraries are not installed as part of Oracle BI Applications 7.9.x. You must download the libraries from the Hibernate Web site.

2. Install the Oracle BI Applications release 7.9.5.1 software. For more information, see [Section 4.4, "Running the Oracle BI Applications Installer on Windows."](#)
3. Install Informatica PowerCenter Server and Client Tools version 8.1.1 SP5 software. For more information, see [Section 4.5, "Installing Informatica PowerCenter Services Software."](#)
4. Restore the prebuilt Oracle BI Repository file. For more information, see [Section 4.6, "Restoring the Prebuilt Informatica Repository."](#)
5. Configure the Informatica PowerCenter components. For more information, see [Section 4.7, "Configuring Informatica PowerCenter Components."](#)
6. Install and Configure the DAC Client. For more information, see [Section 4.9, "Configuring the DAC Client."](#)
7. Import BI Applications metadata into the DAC Repository. For more information, see [Section 4.10.3, "Importing Metadata into the DAC Repository."](#)
8. Create the Oracle Business Analytics Warehouse tables. For more information, see [Section 4.11, "Creating the Oracle Business Analytics Warehouse Tables."](#)

9. Install and configure the DAC Server. For more information, see [Section 4.12, "Installing the DAC Server."](#)
10. Configure DAC Integration Settings. For more information, see [Section 4.14, "Configuring DAC Integration Settings."](#)
11. Create Relational Connections in Informatica Workflow Manager, as specified in [Section 4.15, "Configuring Relational Connections in Informatica Workflow Manager"](#).
12. Configure the SiebelUnicodeDB Custom Property, as specified in [Section 4.16, "Configuring the SiebelUnicodeDB Custom Property"](#).
13. Perform required post-installation tasks (for more information, see [Section 4.18, "Additional Configuration Tasks"](#)).

For additional and supporting configuration tasks, see [Section 4.18, "Additional Configuration Tasks"](#).

---

**Note:** After you have completed the above steps and completed any customizations that you need to make, you are ready to perform a full load of your OLTP data. For an example of performing a full load of OLTP data, see [Section 4.19, "About Running A Full Load ETL."](#) For detailed information about using DAC to perform ETL processes, see *Oracle Business Intelligence Data Warehouse Administration Console Guide*.

---

## 4.3 Mandatory Requirements and Preinstallation Tasks

This section includes mandatory requirements that you must satisfy and preinstallation tasks that you must perform before you can deploy Oracle BI Applications.

This section contains the following topics:

- [Section 4.3.1, "Oracle Business Intelligence Infrastructure Requirements"](#)
- [Section 4.3.2, "Informatica PowerCenter Requirements"](#)
- [Section 4.3.3, "Code Page and Data Movement Requirements"](#)
- [Section 4.3.4, "Preinstallation Tasks"](#)

### 4.3.1 Oracle Business Intelligence Infrastructure Requirements

Installation of Oracle BI Applications requires that you have already installed Oracle Business Intelligence Enterprise Edition on a Windows machine. Oracle Business Intelligence Enterprise Edition is the infrastructure required to run the Oracle BI Applications installer. The instance of Oracle Business Intelligence Enterprise Edition must be a complete installation; that is, you must install it using the Complete setup type option.

**Note:** This instance of Oracle Business Intelligence Enterprise Edition does not need to be the functional version that you will use to deploy reports and dashboards in your live system. This instance is only required to enable the Oracle BI Applications installer to install the Oracle BI Applications files on a machine.

To determine the minimum version of Oracle Business Intelligence Enterprise Edition that is supported for this release of Oracle BI Applications, refer to the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

## 4.3.2 Informatica PowerCenter Requirements

This section contains information about the Oracle BI Applications requirements relating to Informatica PowerCenter.

Oracle BI Applications release 7.9.5.1 requires Informatica PowerCenter 8.1.1 SP5. If you license Informatica PowerCenter with Oracle BI Applications, you install Informatica PowerCenter 8.1.1 SP5 from the Oracle BI Applications installation disk. For more information, see [Section 4.5, "Installing Informatica PowerCenter Services Software."](#)

If you license Informatica PowerCenter separately and you do not have Informatica PowerCenter 8.1.1 SP5, you must upgrade your Informatica PowerCenter license to 8.1.1 SP5 before deploying Oracle BI Applications.

Informatica PowerCenter 8.1.1 has significant architecture changes from previous versions. Before installing Oracle BI Applications, Oracle recommends that you do the following:

- Read the Informatica PowerCenter documentation to familiarize yourself with the new architecture, components, and features. Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_Party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.
- Plan your topology carefully before installing Informatica PowerCenter Services.

You must install Informatica PowerCenter Client Tools on the same machine as the DAC Client and Informatica PowerCenter Services on the same machine as the DAC Server.

Informatica PowerCenter Services runs on 32-bit or 64-bit platforms. For more information see the topic titled 'Working with 32-bit and 64-bit Platforms' in the chapter on PowerCenter Prerequisites in the *Informatica PowerCenter Installation and Configuration Guide*.

### About Changes in Informatica Terminology

This section lists changes in terminology in Informatica PowerCenter 8.1.1 SP5.

**Table 4–1 Changes in Informatica Terminology**

Term used in 7.1.x versions of Informatica PowerCenter	Term used in 8.1.1 version of Informatica PowerCenter
Informatica Repository	Informatica Repository The Informatica Repository is managed by the Repository Service.
Informatica Repository Server	Not applicable. This component does not exist in Informatica PowerCenter 8.1.1. The Informatica Repository is managed by the Repository Service.
Informatica Server	Integration Services
Not applicable	Repository Service

## 4.3.3 Code Page and Data Movement Requirements

The Oracle Business Analytics Warehouse can be deployed in various code page environments and supports global deployments. Data movement in the following source database and data warehouse configuration modes are supported:

- Unicode to Unicode
- Code page (multi- or single-byte) to Unicode
- Code page to Code page (where the code pages are the same)

Oracle BI Applications uses Informatica PowerCenter to perform extract, transform and load routines to move data from source database(s) to the Oracle Business Analytics Warehouse.

During the installation and configuration procedures described in this chapter, you will make various settings to enable accurate data movement. Use the guidelines and references noted below to determine values for these settings that are appropriate for your environment:

- Determining the source to target configuration mode. Consult your database administrator to determine the code page your source OLTP database uses. Based on the type of data that will be moved from one or more source databases to the Oracle Business Analytics Warehouse, determine what code page you will need to use for the Oracle Business Analytics Warehouse database. Consider future requirements for storing data when determining what code page to use for the Oracle Business Analytics Warehouse.

For accurate data movement from source database to target, the code page of the Oracle Business Analytics Warehouse (target) must be a superset of the code page of the source database. Informatica considers a code page to be a superset of another code page when the code page contains all the characters encoded in the other code page and additional characters not encoded in the other code page.

If your source to target configuration mode for data movement is Unicode to Unicode, you will set a custom property called SiebelUnicodeDB on the Integration Services. Configuration modes of code page to Unicode or code page to code page do not require this property to be set.

**Note:** To enable data movement from source(s) to the Oracle Business Analytics Warehouse, you will set relaxed Code Page Validation for the Integration Services. You must ensure that the target code page is a superset of the source code page for accurate data movement.

- Determining the Data Movement Mode. Before you deploy Oracle BI Applications, you must determine what data movement mode to use (ASCII or Unicode) for the PowerCenter Integration Service. The Character Data Movement Mode is an Informatica PowerCenter Integration Service option that you choose based on whether you want to move single-byte or multi-byte data. Choose Unicode if non-ASCII characters have to be moved. Otherwise, choose ASCII. The Data Movement Mode option is configurable and can be reset after installation.

To set the Data Movement Mode, log into Informatica PowerCenter Administration Console, select the Integration Service, then click the Properties tab, then display the General Properties tab, and set the DataMovementMode value.

- Determining code pages for Informatica PowerCenter components. In addition to source and target code pages, Informatica PowerCenter uses code pages for PowerCenter Client, the Integration Service, the Informatica Repository, and PowerCenter command line programs (pmcmd and pmrep, which are used by DAC to communicate with PowerCenter Services). Carefully review the Informatica documentation on 'Understanding Globalization' in the PowerCenter Administration Guide, particularly if your environment requires Data Mode Movement mode to be set to UNICODE. The section discusses code page compatibility and code page requirements for the Informatica components.

- Setting environment variables. The Informatica installer requires the appropriate locale to be set on UNIX machines. Use LANG, LC\_CTYPE or LC\_ALL to set the UNIX code page. The installer uses the locale environment variable to generate the list of compatible code pages. For more information, see the topic titled 'Locale Environment Variables' in the *Informatica PowerCenter Installation and Configuration Guide*.

If your environment uses Oracle or DB2 database, you need to set environment variables NLS\_LANG or DB2CODEPAGE. For information on how to set these environment variables see [Section 4.3.4.2.1, "How to Set the NLS\\_LANG Environment Variable for Oracle Databases"](#) and [Section 4.3.4.2.2, "How to Set the DB2CODEPAGE Environment Variable for DB2 Databases."](#)

- Configuring Relational Connections. When you configure relational connections in the Workflow Manager, choose a code page that is compatible with the code page of the database client. If you set a database environment variable to specify the language for the database, ensure the code page for the connection is compatible with the language set for the variable. For example, if you set the NLS\_LANG environment variable for an Oracle database, ensure that the code page of the Oracle connection is identical to the value set in the NLS\_LANG variable.

For more information about data movement modes, refer to the Informatica PowerCenter documentation. Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_Party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.

## 4.3.4 Preinstallation Tasks

This section explains the mandatory tasks that you must perform for an Oracle BI Applications deployment. It contains the following topics:

- [Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"](#)
- [Section 4.3.4.2, "Install Database Connectivity Software"](#)
- [Section 4.3.4.3, "Create ODBC Database Connections"](#)
- [Section 4.3.4.4, "Set Environment Variables"](#)
- [Section 4.3.4.5, "Download and Install the Java JDK"](#)
- [Section 4.3.4.6, "Download Hibernate Libraries"](#)

### 4.3.4.1 Create Databases for Oracle BI Applications and Informatica PowerCenter Components

Before you install Oracle BI Applications and Informatica PowerCenter, use your target database tool to create database instances to hold the following:

- DAC Repository
- the PowerCenter domain configuration
- Oracle Business Analytics Warehouse

**Note:** Make sure that the Oracle Business Analytics Warehouse instance is granted the SSE\_ROLE. For more information, see [Section 4.3.4.1.1, "How to Create the SSE Role."](#)

You can store the DAC Repository and the Oracle Business Analytics Warehouse in separate databases or in the same database.



For information about supported database platforms, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. The Informatica and DAC repositories are not supported on DB2/390 or Teradata databases. If your source or Oracle Business Analytics Warehouse database is DB2/390 or your Oracle Business Analytics Warehouse database is Teradata, you need to use a DB2-UDB, MS SQL Server, or Oracle database for both the DAC Repository and the Informatica Repository.

For details on creating the Informatica Repository and PowerCenter domain configuration databases, see the section on creating databases in the *Informatica PowerCenter Installation and Configuration Guide*. Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_Party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.

**4.3.4.1.1 How to Create the SSE Role** Follow this procedure to create the SSE role for the Oracle Business Analytics Warehouse database.

#### To create the SSE Role

1. Create a database role named SSE\_ROLE (SSEROLE for DB2/390 databases). Assign this role to the database user. For instructions on creating roles, see the documentation provided with your database.

Note the following:

- For an Oracle database, when you create the SSE\_ROLE role, you need to grant the following privileges:
  - CONNECT
  - RESOURCE
- For an Oracle database, if the DAC Repository and the Informatica Repository are stored in a different database from the data warehouse database, the SSE\_ROLE must have the following additional privileges:
  - READ
  - WRITE
  - UPDATE
  - DELETE
  - GRANT
- If you are using a Teradata database, you do not need to create the SSE role.
- If you are using a DB2/390 database, you can use the alternative SSE role name SSEROLE if required. To specify a different role name, modify the 'set GRANTEE="SSEROLE"' line in the OracleBI\dwrep\createtables.bat file.

#### 4.3.4.2 Install Database Connectivity Software

You must install and configure the appropriate database native connectivity client software on the machines that will host the Informatica PowerCenter Services, DAC Server, and DAC Client.

The machines that will host these components require connectivity to the Oracle Business Analytics Warehouse (target) database, transactional (source) database(s), and DAC Repository and Informatica Repository databases.

**Notes**

- For instructions on how to install and configure the native database client connectivity software, refer to the *Informatica PowerCenter Installation and Configuration Guide*:
- To install and configure on the Windows machine that will host the DAC Client and PowerCenter Tools, see section 'Connecting to Databases from Windows' in the Informatica Installation and Configuration Guide. Also refer to this section if the DAC Server and PowerCenter Services will be installed on Windows.
- To install and configure on a UNIX or Linux machine that will host the DAC Server and PowerCenter Services, see section 'Connecting to Databases from UNIX' in the *Informatica PowerCenter Installation and Configuration Guide*.
- Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_Party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.
- For Oracle databases, set the NLS\_LANG environment variable on the machines where the Oracle client was installed, as described in [Section 4.3.4.2.1, "How to Set the NLS\\_LANG Environment Variable for Oracle Databases"](#).
- For DB2 databases, set the DB2CODEPAGE environment variable on the machines where the DB2 client was installed, as described in [Section 4.3.4.2.2, "How to Set the DB2CODEPAGE Environment Variable for DB2 Databases"](#).

**4.3.4.2.1 How to Set the NLS\_LANG Environment Variable for Oracle Databases** Follow this procedure to set the NLS\_LANG environment variable for Oracle databases.

---

---

**Note:** You need to set the NLS\_LANG environment variable on each machine that has the Oracle client installed.

---

---

**To set the NLS\_LANG environment variable for Oracle databases**

1. Determine the NLS\_LANG value.
  - a. In the data warehouse database, run the following command:

```
SELECT * FROM V$NLS_PARAMETERS
```
  - b. Make a note of the NLS\_LANG value, which is in the format [NLS\_LANGUAGE]\_[NLS\_TERRITORY].[NLS\_CHARACTERSET].  
For example: `American_America.UTF8`
2. For Windows:
  - a. Navigate to Control Panel > System and click the Advanced tab. Click Environment Variables.
  - b. In System variables section, click New.
  - c. In the Variable Name field, enter NLS\_LANG.
  - d. In the Variable Value field, enter the NLS\_LANG value that was returned in Step 1.

The format for the NLS\_LANG value should be [NLS\_LANGUAGE]\_[NLS\_TERRITORY].[NLS\_CHARACTERSET].

For example: `American_America.UTF8`.

---



---

**Note:** The NLS\_LANG character set should reflect the setting of the operating system character set of the client. For example, if the database character set is AL32UTF8 and the client is running on a Windows operating system, then you should not set AL32UTF8 as the client character set in the NLS\_LANG parameter because there are no UTF-8 WIN32 clients. Instead, the NLS\_LANG setting should reflect the code page of the client. For example, on an English Windows client, the code page is 1252. An appropriate setting for NLS\_LANG is AMERICAN\_AMERICA.WE8MSWIN1252.

Setting NLS\_LANG correctly allows proper conversion from the client operating system character set to the database character set. When these settings are the same, Oracle assumes that the data being sent or received is encoded in the same character set as the database character set, so character set validation or conversion may not be performed. This can lead to corrupt data if the client code page and the database character set are different and conversions are necessary

---



---

3. For UNIX and Linux, set the variable as shown below:

```
setenv NLS_LANG <NLS_LANG>
```

For example: `setenv NLS_LANG American_America.UTF8.`

If your data is 7-bit or 8-bit ASCII and the Integration Service is running on UNIX or Linux, then set `NLS_LANG <NLS_LANGUAGE>_<NLS_TERRITORY>.WE8ISO8859P1`

---



---

**Caution:** Make sure you set the NLS\_LANG variable correctly, as stated in this procedure, or your data will not display correctly.

---



---

**4.3.4.2.2 How to Set the DB2CODEPAGE Environment Variable for DB2 Databases** Follow this procedure to set the DB2CODEPAGE environment variable for DB2 databases on machines that have the DB2 client installed.

**To set the DB2CODEPAGE environment variable for DB2 databases**

1. Determine the DB2CODEPAGE value.
  - a. Connect to the Source database, using the following command:
 

```
SELECT CODEPAGE FROM SYSCAT.DATATYPES WHERE TYPENAME = 'VARCHAR'
```
  - b. Make a note of the result.
 

For example: 1208
2. For Windows:
  - a. Navigate to Control Panel > System and click the Advanced tab. Click Environment Variables.
  - b. In System variables section, click New.
  - c. In the Variable Name field, DB2CODEPAGE.
  - d. In the Variable Value field, enter the value that was returned in Step 1.
3. For UNIX and Linux, set the variable as shown below:

```
setenv DB2CODEPAGE <DB2CODEPAGE value>
```

For example: `setenv 1208`.

#### 4.3.4.3 Create ODBC Database Connections

The DAC Client uses an ODBC connection to create and alter tables in the Oracle Business Analytics Warehouse. Create an ODBC connection to the Oracle Business Analytics Warehouse database on the machine that hosts the DAC Client, as described below.

Additionally, if your source system is Siebel CRM, you must create an ODBC connection to the OLTP (source) database on the machine where you will install Oracle BI Applications software. This ODBC connection will be used by the `ddlmp` utility.

For the ODBC connections to Oracle databases, you must use the Oracle Merant ODBC Driver that is installed with the Oracle BI Applications installer. For all other databases, you should use ODBC drivers supplied by your database vendor.

Refer to the appropriate instructions for your database type:

- [Section 4.3.4.3.1, "How to Create ODBC Connections for DB2 Databases"](#)
- [Section 4.3.4.3.2, "How to Create ODBC Connections for Oracle Databases"](#)
- [Section 4.3.4.3.3, "How to Create ODBC Connections for SQL Server Databases"](#)
- [Section 4.3.4.3.4, "How to Create ODBC Connections for Teradata Databases"](#)

---

---

**Note:** For information about creating database connections for Informatica PowerCenter, refer to the Informatica documentation.

---

---

**4.3.4.3.1 How to Create ODBC Connections for DB2 Databases** Follow these instructions for creating ODBC connections for DB2 databases on Windows. For instructions on creating ODBC connections for DB2 databases on UNIX or Linux, see the documentation provided with your database.

##### To create ODBC connections for DB2 databases on Windows

1. Using the DB2 Client Configuration Assistant, create a database connection to the Oracle Business Analytics Warehouse database on the machine that will host the DAC Client. If your source system is Siebel CRM, also create an ODBC connection to the transactional database on the machine where you will run the Oracle BI Applications installer.

**Note:** If you use the DB2 Client Configuration Assistant to create database connections, you can omit step 2, because the DB2 Client Configuration Assistant automatically creates System DSNs (default behavior).

2. If necessary, on Windows, in the System DSN tab of the ODBC Data Source Administrator, create an ODBC connection to the Oracle Business Analytics Warehouse (and, if your source system is Siebel CRM, to the transactional database) using an ODBC driver.
3. Test the connections to make sure they work.

**4.3.4.3.2 How to Create ODBC Connections for Oracle Databases** Follow these instructions for creating ODBC connections for Oracle databases on Windows. For instructions on creating ODBC connections for Oracle databases on UNIX or Linux, see the documentation provided with your database.

**Note:** You must use the Oracle Merant ODBC driver to create the ODBC connections. The Oracle Merant ODBC driver is installed by the Oracle BI Applications installer. Therefore, you will need to create the ODBC connections after you have run the Oracle BI Applications installer and have installed the DAC Client.

#### To create ODBC connections for Oracle databases on Windows

1. On the machine that will host the DAC Client, use the System DSN tab of the ODBC Data Source Administrator to create an ODBC connection to the Oracle Business Analytics Warehouse database using the Oracle Merant ODBC driver that is supplied with Oracle BI Applications.

For example, you might create a database connection called Connect\_to\_OLAP.

2. (If your source system is Siebel CRM) In Windows, in the System DSN tab of the ODBC Data Source Administrator, create an ODBC connection to the Siebel transactional database using the Oracle Merant Closed driver.
3. Test the connections to make sure they work.

**4.3.4.3.3 How to Create ODBC Connections for SQL Server Databases** Follow these instructions for creating ODBC connections for SQL Server databases on Windows. For instructions on creating ODBC connections for SQL Server databases on UNIX or Linux, see the documentation provided with your database.

#### To create ODBC connections for SQL Server databases on Windows

1. In Windows, in the System DSN tab of the ODBC Data Source Administrator, create the following:
  - An ODBC connection to the Oracle Business Analytics Warehouse database on the machine that will host the DAC Client.
  - (If your source system is Siebel CRM) An ODBC connection to the transactional database on the machine where you will run the Oracle BI Applications installer.

**Note:** Select SQL Server as the ODBC driver.

2. Test the connections to make sure they work.

**Note:** When you use the ODBC Data Source Administrator to create a database connection, make sure that you select the SQL Server authentication option using a login ID and password entered by the user.

**4.3.4.3.4 How to Create ODBC Connections for Teradata Databases** Follow these instructions for creating ODBC connections for Teradata databases.

#### To create ODBC connections for Teradata databases on Windows

1. On the Windows machine that will host the DAC Client, in the System DSN tab of the ODBC Data Source Administrator, create an ODBC connection to the Oracle Business Analytics Warehouse Teradata database.

2. Set the following parameters:

Field=Enter

DateFormat=AAA

SessionMode=ANSI

NoScan=Yes

3. For Unicode environments, in the Teradata ODBC Driver Advanced Options dialog box, set the Character Set parameter to UTF8.

#### To create ODBC connections for Teradata databases on UNIX

1. Using the Teradata ODBC driver, create an ODBC connection for the Teradata database.
2. Set the following variables in the ODBC.INI file:
  - DateFormat=AAA
  - SessionMode=ANSI
  - NoScan=Yes
3. For UNICODE environments, in the Teradata ODBC Driver Advanced Options dialog box, add the following:
  - CharacterSet=UTF8

#### 4.3.4.4 Set Environment Variables

Before you run the Informatica PowerCenter installer, you must set environment variables as described in 'Step 5. Configure Environment Variables' in the chapter 'Before You Install' in the *Informatica PowerCenter Installation and Configuration Guide*. For UNIX machines, you must set environment variables for database client, Library Path and Locale as described. Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.

#### 4.3.4.5 Download and Install the Java JDK

The DAC Client and DAC Server require Java JDK. Install Java JDK on the machines that will host the DAC Client and DAC Server. On Windows, you must install JDK with no spaces in the directory path.

If required, you can have other versions of SDK installed on the same server to support other applications. For example, you might have Java SDK version 1.5 installed for the DAC, and Java SDK version 1.4.2 installed for a different application that is not supported on Java SDK version 1.5.

---

---

**Note:** For information about which version of the Java SDK is supported, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

---

---

#### To install Java SDK

1. Download and install the Java JDK for your operating system(s) from a Web site that provides Java JDK downloads (for example, the Sun or IBM Web site). You must download and install the Java JDK for either 32 bit or 64 bit depending on your operating system.

---

---

**Note:** On Windows, make sure that there are no spaces in the JDK directory path. If there are spaces in the directory path, DAC will not start (the \OracleBI\DAC\startclient.bat file will fail).

---

---

#### 4.3.4.6 Download Hibernate Libraries

To run the DAC Client or DAC Server, you need to have libraries from an open source software product called Hibernate. Hibernate libraries are not installed as part of Oracle BI Applications 7.9.x. You must download the libraries from the Hibernate Web site.

Oracle recommends that you download Hibernate Core Package version 3.2.x GA or later. Newer versions of Hibernate Core Package 3.2 are now generally available (for example, Hibernate Core Package version 3.2.5 GA). DAC is supported on the libraries of these versions also. You can download Hibernate Core Package from <http://www.hibernate.org>.

##### To download the Hibernate libraries

1. Go to the Web site <http://hibernate.org>.
2. Navigate to the Download area and download Hibernate Core Package version 3.2.x. GA to a local machine.
3. Unzip the files and directories in the hibernate- $\langle 3.2.x \rangle$ .ga.zip file (for example, hibernate-3.2.2.ga.zip).

The files will unzip into a directory named hibernate-3.2. As part of the installation and set up of DAC described later in this chapter, you will copy the Hibernate libraries that you downloaded to the DAC Client machine.

## 4.4 Running the Oracle BI Applications Installer on Windows

The Oracle BI Applications installer runs on Windows and requires the Oracle Business Intelligence infrastructure to be installed. For more information on Oracle Business Intelligence infrastructure requirements, see [Section 4.3.1, "Oracle Business Intelligence Infrastructure Requirements"](#).

This section explains how to install the Oracle BI Applications files using the Oracle BI Applications Installation Wizard.

When you run the Oracle BI Applications Installation Wizard, the Oracle BI Applications files are installed into the existing Oracle Business Intelligence infrastructure directory (for example,  $\langle \text{DRIVE} \rangle : \backslash \text{OracleBI} \backslash$ ).

---

**Note:** If you have a previous release of Oracle BI Applications installed, you must uninstall this release before you run the installer for Oracle BI Applications release 7.9.5.1. If you do not uninstall the previous version, some release 7.9.5.1 directories will not be correctly installed. (Make a back-up of your Oracle BI Repository, Presentation Catalog, and DAC and dwrep directories before you uninstall the previous release.)

---

##### To install Oracle BI Applications on Windows

1. Access the installation files on the installation CD-ROM, and then run the program setup.exe to display the Welcome page.

---

**Note:** To run the installer in console (or text) mode, run the command setup.exe -console. You do not see the following screens in console installation mode. Instead, you enter input as plain text in the terminal window when prompted.

---

2. In the Welcome page, click Next.
3. Enter or browse for the location for the Oracle Business Intelligence infrastructure (for example, <DRIVE>:\OracleBI\) and for the Oracle Business Intelligence data (for example, <DRIVE>:\OracleBIData\). Click Next.

4. Select the Oracle BI Applications that you want to install, and then click Next.

To determine the applications you should install for the Oracle BI Applications licenses you have purchased, see the spreadsheet entitled '2-Installer Options' in the Oracle Business Intelligence Applications Licensing and Packaging Guide. This guide is part of the Oracle Business Intelligence Media Pack.

Once you have installed applications, you can deploy them for one or more source systems selectively when you create the DAC Repository in a later step. For more information, see [Section 4.10.3, "Importing Metadata into the DAC Repository."](#)

5. At the summary page, review the summary information, and click Next to start the installation.

The installer installs the Oracle BI Applications directories and files in the Oracle Business Intelligence infrastructure installation.

---

---

**Note:** Even when the progress bar on the installer reports 100% complete, you must wait until the **Finish** button is displayed.

---

---

6. Click Finish.

**Tip:** Look in the <DRIVE>:\OracleBI\Document\version\_apps.txt file in the Oracle Business Intelligence infrastructure directory to check that you have installed the correct version of Oracle BI Applications.

When the installation is complete, verify that the following directories or files are installed:

- The DAC folder.
- The dwrep folder.
- The Oracle BI Applications repository file named OracleBIAnalyticsApps.rpd in the OracleBI\server\Repository folder.
- The Presentation Catalog in the OracleBIData\web\catalog folder. The Presentation Catalog is provided as a zipped file named EnterpriseBusinessAnalytics.zip and will need to be unzipped before it can be used.
  - Language files (for the languages available for Oracle BI Applications) in OracleBIData\web\res folder.
  - Seed Data folder containing localization files.
  - Upgrade folder containing repository files from previous versions of Oracle BI Application and Siebel Analytics Application releases.

#### Notes

- The credentials required to log into the OracleBIAnalyticsApps.rpd file using the Oracle Business Intelligence Administration Tool are
  - Username: Administrator



- Password: SADMIN
- For information on configuring the Oracle BI Repository, see the *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*.
- Refer to the Oracle Business Intelligence Enterprise Edition documentation for more information on working with the Oracle BI Repository and Presentation Catalog.

## 4.5 Installing Informatica PowerCenter Services Software

This section explains how to install Informatica PowerCenter Services and Client. It contains the following topics:

- [Section 4.5.1, "Installing Informatica PowerCenter Client Tools \(Windows\)"](#)
- [Section 4.5.2, "Installing Informatica PowerCenter Services"](#)

### Notes

- You need to determine the machines that will host Informatica PowerCenter Services and Informatica PowerCenter Client Tools. The PowerCenter Services can be installed on UNIX or on Windows. The PowerCenter Client Tools must be installed on Windows.

Oracle recommends that you:

- Co-locate the DAC Client with the PowerCenter Client Tools.
- Co-locate the DAC Server with the PowerCenter Services.

DAC produces parameter files that are used by Informatica. If an execution plan fails in DAC and you want to debug the workflow by running it directly from Informatica, then the parameter file produced by DAC should be visible to Informatica.

- You must meet all requirements and complete preinstallation tasks as described in [Section 4.3, "Mandatory Requirements and Preinstallation Tasks"](#) and in the chapter 'Before You Install' of the *Informatica PowerCenter Installation and Configuration Guide*. Informatica PowerCenter 8.1.1 SP5 documentation is provided in the Third\_Party\_Software\Informatica folder of the Oracle BI Applications release 7.9.5.1 installation disk.
- When you install PowerCenter Services and Client Tools version 8.1.1 SP5, use the Informatica PowerCenter installation documentation in conjunction with the steps provided in the following sections. The *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users* summarizes the steps and assumes a single-machine installation for the PowerCenter Services components.

**Note:** For a new installation, you must select the 'Create new PowerCenter Repository' option on the PowerCenter Repository Configuration screen.

---

**Note:** If you have licensed Informatica PowerCenter separately from Oracle BI Applications, skip this section and go to [Section 4.6, "Restoring the Prebuilt Informatica Repository."](#)

Make sure that you have installed the required version of Informatica PowerCenter (for more information, see [Section 4.3.2, "Informatica PowerCenter Requirements"](#)).

---

## 4.5.1 Installing Informatica PowerCenter Client Tools (Windows)

This section explains how to install Informatica PowerCenter 8.1.1 SP5 Client Tools for an Oracle BI Applications deployment. For detailed generic information about installing Informatica 8.1.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, and related documentation.

**Note:** Oracle recommends that you co-locate the PowerCenter Client Tools with the the DAC Client.

Follow this procedure to install the Informatica PowerCenter Client Tools for an Oracle BI Applications deployment.

### To install Informatica PowerCenter Client Tools on Windows

1. Run the program `install.exe` from the `\Third_Party_Software\Informatica\PowerCenter_8.1.1_SE_for_Windows_<bit mode>` folder on the Oracle BI Applications installation DVD.  
  
Alternatively, run `install.bat` from the `\Third_Party_Software\Informatica\PowerCenter_8.1.1_SE_for_<OS>\Client\Disk1` folder.
2. At the 'Please select an option to install' dialog box, click Client.
3. Follow the instructions on the Informatica PowerCenter Client 8.1.1 installation wizard.
4. Install Informatica PowerCenter SP5 patch from the `\Third_Party_Software\Informatica\PowerCenter_8.1.1_SP5_for_Windows_<bit mode>` folder, as follows:
  - Install Client - accept all default installation settings.

## 4.5.2 Installing Informatica PowerCenter Services

This section explains how to install Informatica PowerCenter Services 8.1.1 SP5 for an Oracle BI Applications deployment. For detailed generic information about installing Informatica 8.1.1, refer to the *Informatica PowerCenter Installation and Configuration Guide* and related documentation.

Note the following:

- Oracle recommends that you co-locate the PowerCenter Services with the DAC Server.
- During the Informatica PowerCenter Services installation, you must select the 'Create new PowerCenter Repository' option on the PowerCenter Repository Configuration screen.

### To install Informatica PowerCenter Services 8.11 SP5

1. Run the program `install.exe` from the `\Third_Party_Software\Informatica\PowerCenter_8.1.1_SE_for_Windows_<bit mode>` folder on the Oracle BI Applications installation DVD.

Alternatively, run `install.bat` from the `\Third_Party_Software\Informatica\PowerCenter_8.1.1_SE_for_<OS>\Client\Disk1` folder.

To start the installer on UNIX or Linux, run `install.sh` at a shell command line from the DVD root directory or the root directory in the location where you downloaded the installer.

On UNIX and Linux, the PowerCenter installer checks the operating system and verifies that the environment variables are correctly set. The environment

variables must be specifically set to enable the PowerCenter installer to run processes and connect to databases and services. If the environment variables are not set, the PowerCenter installer gives you the option to stop the installation and modify the environment variables.

On UNIX or Linux, select the option to use the GUI mode installation.

2. At the 'Please select an option to install' dialog box, click Server.
3. Follow the instructions on the Informatica PowerCenter Services installation wizard, as shown in the following table.

---

**Note:** The installation wizard pages that you see depend on the licensing options that you have purchased. For example, the Application Server Configuration page is only displayed if you have purchased the PowerCenter Data Analyzer option.

---

Page	Your Action	Notes
Welcome	Click Next.	
License Key	Click Browse, then locate and select the license key text file.	If you have purchased the 'Informatica OEM PowerCenter ETL Server and PowerConnect Adapters' license from Oracle, use the Oracle_All_OS_Prod.key license key file. This file is located in the DVD root directory of the Informatica PowerCenter DVD that is provided with Oracle BI Applications.
Installation Prerequisites	Click Next.	
Installation Set	Select the <b>Install PowerCenter 8.1.1</b> radio button.	For information about upgrading to Informatica PowerCenter 8.1.1, refer to <i>Informatica PowerCenter Installation and Configuration Guide</i> .
DA & MM License Key	Not applicable.	
Installation directory	Specify the installation folder.	On Windows, do not install Informatica in a directory that has a space in the directory name. For example, do not install in D:\Program Files\INFA. If you do install into a directory with a space in the name, you will cause errors in DAC.
Application Server Configuration	Not applicable.	
Pre-Installation Summary	Click Install.	
Installing...	Not applicable.	
Domain Configuration	At the Create or Join Domain page, select the <b>Create New Domain</b> radio button.	

Page	Your Action	Notes
Domain Configuration	At the Configure Domain Database page, specify the database details.	<p>Configure Domain Database page:</p> <ul style="list-style-type: none"> <li>Specify the database details for the database account that stores the PowerCenter domain configuration metadata. Use the database that you created earlier (for more information, see <a href="#">Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"</a>). You can use the same database that is used to store the Informatica Repository.</li> </ul>
Domain Configuration	At the Configure Domain page, specify the domain details.	<p>Configure Domain page:</p> <ul style="list-style-type: none"> <li>Informatica PowerCenter creates a default domain called Domain_&lt;install machine name&gt; and a default node called Node01_&lt;install machine name&gt;.</li> <li>Specify a domain password (for example, admin), which you use to log into PowerCenter Administration Console.</li> <li>Make sure that you select the <b>Create Repository Service</b> check box.</li> </ul>
Node Configuration	At the PowerCenter Repository Configuration page, select the <b>Create new PowerCenter repository</b> radio button, and specify the Services details. You must specify the appropriate repository Code Page setting.	<p>PowerCenter Repository Configuration page:</p> <ul style="list-style-type: none"> <li>Specify the database details that you created for the Informatica Repository.</li> <li>Choose a code page that is compatible with the code page of the machine where you install PowerCenter. For more information, refer to the Informatica PowerCenter documentation, and <a href="#">Section 4.3.3, "Code Page and Data Movement Requirements."</a></li> </ul>
Node Configuration	<p>At the Configure Integration Service page, select the <b>Create Integration Service</b> check box.</p> <p>Select the data character movement mode, either ASCII or Unicode.</p>	<p>PowerCenter Services Configuration page:</p> <ul style="list-style-type: none"> <li>The username here is the username for the Informatica Repository (for example, Administrator).</li> <li>The password here is the password for the Informatica Repository (for example, Administrator).</li> <li>Select Unicode if non-ASCII characters are to be moved from source database to the Oracle Business Analytics Warehouse. Otherwise, select ASCII. For more information, see <a href="#">Section 4.3.3, "Code Page and Data Movement Requirements."</a></li> </ul>
Authentication	Not applicable.	
Repository Configuration	Not applicable.	

Page	Your Action	Notes
Post-Installation Summary	Click Done.	
	4. Stop any Informatica services that are running.	
	5. Install the Informatica PowerCenter SP5 patch from the \Third_Party_Software\Informatica\PowerCenter_8.1.1_SP5_<OS>_<bit mode> folder, as follows:	
	<ul style="list-style-type: none"> <li>■ Install Server - select the Custom installation mode, and on the PowerCenter Components page, select PowerCenter Services 8.1.1 SP5, and deselect PowerCenter Data Analyzer 8.1.1 SP5 and PowerCenter Metadata Manager 8.1.1 SP5. Accept the other default installation settings.</li> </ul>	

## 4.6 Restoring the Prebuilt Informatica Repository

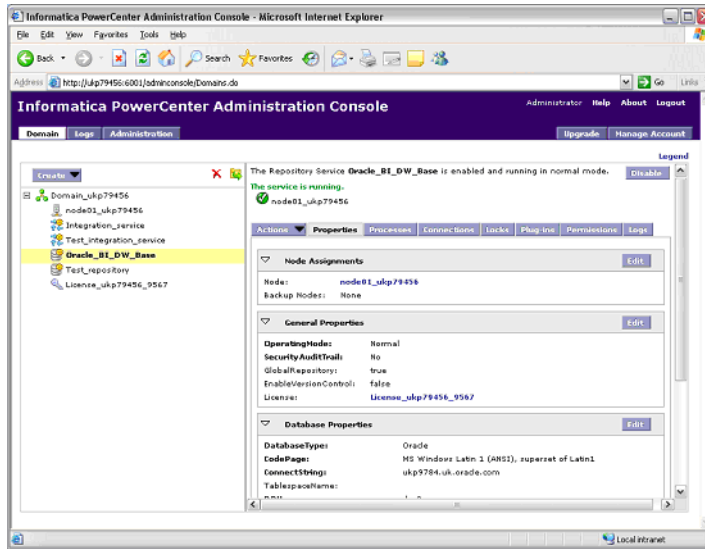
An Informatica Repository file called Oracle\_BI\_DW\_Base.rep is installed into the OracleBI\dwrep\Informatica\Repository directory during the Oracle BI Applications installation. For use with deployments where the Oracle Business Analytics Warehouse is on Teradata, a repository file called Oracle\_BI\_DW\_Teradata.rep is installed.

You use the Restore option in Informatica PowerCenter Administration Console to load the prebuilt Oracle\_BI\_DW\_Base repository or Oracle\_BI\_DW\_Teradata repository (in a Teradata environment).

### To load the prebuilt Oracle\_BI\_DW\_Base.rep or Oracle\_BI\_DW\_Teradata.rep repository into Informatica

1. Copy the file Oracle\_BI\_DW\_Base.rep (or Oracle\_BI\_DW\_Teradata.rep if your Oracle Business Analytics Warehouse is on Teradata) from the OracleBI\dwrep\Informatica\Repository directory on the machine where the Oracle BI Applications installer was run to the following folder on the machine where Informatica PowerCenter Services has been installed:
  - On Windows copy the file to the \Informatica PowerCenter8.1.1\server\infa\_shared\Backup directory.
  - On UNIX or Linux copy the file to \$Informatica/PowerCenter8.1.1/server/infa\_shared/Backup directory.
2. In Informatica PowerCenter Administration Console, select the Repository Service that was created during the installation process (for example, Oracle\_BI\_DW\_Base or Oracle\_BI\_DW\_Teradata).

For information about how to connect to Informatica PowerCenter Administration Console, see [Section A.4, "How to Log Into Informatica PowerCenter Administration Console."](#)



3. In the Properties tab, click Edit in the General properties area.
4. Make sure that the OperatingMode value is Exclusive.
5. Click OK.
6. Choose Actions, then Delete Contents.
7. At the Delete contents for <repository name> dialog box, enter the repository username and password (for example, Administrator\Administrator), then click OK.
8. Choose Actions > Restore Contents.
9. At the Restore Contents dialog box, select Oracle\_BI\_DW\_Base (or Oracle\_BI\_DW\_Teradata for Teradata installations) from the Select backup file drop down list.
10. Select the Restore as new check box.
11. Click OK to start the restore process.
12. When the restore process is complete, click Close (or click Save first to save the logging information).

When the restore is complete (the process typically takes approximately ten to twenty minutes), you will see a 'Success' message.

When a repository is restored, the repository becomes a standalone repository. After restoring the repository, you need to promote it to a global repository.

13. In the Properties tab, click Edit in the General properties area.
14. Change the OperatingMode value to Normal.
15. If prompted, enter the repository username and password.

---

**Note:** The prebuilt Oracle\_BI\_DW\_Base and Oracle\_BI\_DW\_Teradata repository files are versioned from Oracle BI Applications release 7.9.3 and higher. For more information about the versioned repositories, see [Appendix J, "About the Versioned Informatica Repository."](#)

---

## 4.7 Configuring Informatica PowerCenter Components

This section explains how to configure Informatica PowerCenter Services for use with Oracle BI Applications. It contains the following topics:

- [Section 4.7.1, "Copying Source Files and Lookup Files"](#)
- [Section 4.7.2, "Setting PowerCenter Integration Services Relaxed Code Page Validation"](#)
- [Section 4.7.3, "Setting PowerCenter Integration Services Custom Properties"](#)

### 4.7.1 Copying Source Files and Lookup Files

You need to copy source files and lookup files from the Oracle BI Applications installation directory to the Informatica directory on the Informatica PowerCenter Services machine.

**Note:** The following instructions assume the Informatica default directory locations for source and lookup files.

#### To copy source files and lookup files to Informatica PowerCenter Services on Windows

1. Copy the source files in `\OracleBI\dwrep\Informatica\SrcFiles` on the machine where the Oracle BI Applications installer was run to `\Informatica PowerCenter8.1.1\server\infa_shared\SrcFiles` on the Informatica PowerCenter Services machine.
2. Copy the lookup files in `\OracleBI\dwrep\Informatica\LkpFiles` on the machine where the Oracle BI Applications installer was run to `Informatica PowerCenter8.1.1\server\infa_shared\LkpFiles` on the Informatica PowerCenter Services machine.

#### To copy source files and lookup files to Informatica PowerCenter Services on UNIX or Linux

1. Copy the source files in `\OracleBI\dwrep\Informatica\SrcFiles` on the Windows machine where the Oracle BI Applications installer was run to `$Informatica/PowerCenter8.1.1/server/infa_shared/SrcFiles` on the Informatica PowerCenter Services machine.
2. Copy the lookup files in `\OracleBI\dwrep\Informatica\LkpFiles` on the Windows machine where the Oracle BI Applications installer was run to `$Informatica/PowerCenter8.1.1/server/infa_shared/LkpFiles` on the Informatica PowerCenter Services machine.

### 4.7.2 Setting PowerCenter Integration Services Relaxed Code Page Validation

PowerCenter Integration Services must be configured for relaxed code page validation.

#### To configure Informatica PowerCenter Integration Services for relaxed code page validation

1. Log in to Informatica PowerCenter Administration Console.
2. Select the Integration Service.
3. Select the Properties tab, then Configuration Properties, and then Edit.
4. Change the value for the property 'ValidateDataCodePages' from 'Yes' to 'No'.

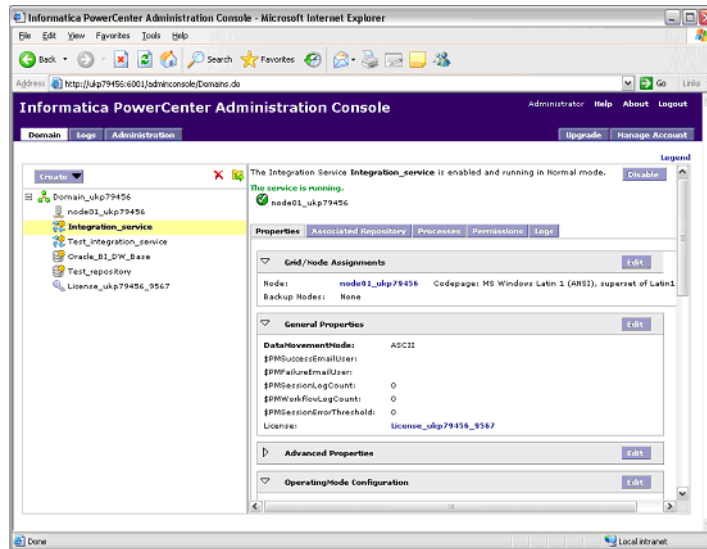
### 4.7.3 Setting PowerCenter Integration Services Custom Properties

Follow this procedure to set PowerCenter Integration Services custom properties.

#### To set Informatica PowerCenter Integration Services Custom Properties

1. In Informatica PowerCenter Administration Console, select the Integration Service.

For example, on the machine where Informatica PowerCenter Services is installed, select Programs, then Informatica PowerCenter 8.1.1, then Services, then Launch Admin Console.



2. Click the Properties tab.
3. In the Custom Properties area, click Edit.
4. Create a custom property by clicking Add to display new Name and Value fields.
5. For each of the custom properties in the table below, click Add and add a new custom property with an appropriate value:

**Table 4–2 Custom Properties for Informatica PowerCenter Integration Services**

Custom Properties Name	Custom Properties Value	Notes
ServerPort	<Server Port Number>	For <Server port number>, enter the number of an available port. For example, 4006. This custom property configures Informatica PowerCenter Services to listen on <Server port number>. DAC communicates with the PowerCenter Integration Services service using this port.
overrideMpltVarWithMapVar	Yes	Enables Informatica to evaluate parameters within mapplets.
DisableDB2BulkMode	Yes	Add this custom property and set value to Yes if your Oracle Business Analytics Warehouse is on DB2/390 database.

6. Click OK to save the details.



7. Make sure that the Integration Service and Repository Service that you created during the Informatica PowerCenter installation are running.

## 4.8 Installing an Additional Instance of the DAC Client on Windows

The DAC Client is installed by the Oracle BI Applications installer. The DAC Client can only be installed on Windows. So, if you performed the steps in [Section 4.4, "Running the Oracle BI Applications Installer on Windows,"](#) you have already installed the DAC Client on the machine where you ran the Oracle BI Applications installer. **Note:** The DAC Server is also installed by the Oracle BI Applications installer at the same time as the DAC Client.

This section contains instructions for installing an additional instance of the DAC Client on a different machine from the one on which you ran the Oracle BI Applications installer.

**Note:** Oracle recommends that you install the DAC Client on the Windows machine that runs the Informatica PowerCenter Client tools.

You must satisfy the following requirements on the machine on which you want to install an additional DAC Client:

- JDK 1.5 or higher must be installed. For more information, see [Section 4.3.4.5, "Download and Install the Java JDK"](#).
- The appropriate ODBC database connection must be created on the machine. For more information, see [Section 4.3.4.3, "Create ODBC Database Connections"](#).
- Download the Hibernate files from the Hibernate Web site. For more information, see [Section 4.3.4.6, "Download Hibernate Libraries"](#).

### To install an additional instance of the DAC Client on Windows

1. Copy the \OracleBI\DAC\ directory and its subfolders from the Oracle BI Applications installation (on Windows) to the target Windows machine (for example, to a directory called \DAC\).
2. Copy the \OracleBI\dwrep\ directory and its subfolders from the Oracle BI Applications installation (on Windows) to the target Windows machine.

**Note:** Alternatively, if the target machine already has Oracle Business Intelligence Enterprise Edition installed, you can run the Oracle BI Applications installer on the target machine to install the DAC Client. For information about how to run the Oracle BI Applications installer, see [Section 4.4, "Running the Oracle BI Applications Installer on Windows."](#)

## 4.9 Configuring the DAC Client

This section contains instructions for configuring the DAC Client.

**Note:** You do not need Oracle Business Intelligence Enterprise Edition installed on a machine to run a DAC Client.

This section contains the following topics:

- [Section 4.9.1, "Copying Hibernate Files to the DAC Directories"](#)
- [Section 4.9.2, "Installing JDBC Drivers for DAC Database Connectivity"](#)
- [Section 4.9.3, "Configuring the DAC Config.bat File"](#)
- [Section 4.9.4, "Enabling DAC Client Communication with Informatica PowerCenter"](#)

## 4.9.1 Copying Hibernate Files to the DAC Directories

Follow these instructions to copy Hibernate files to the appropriate DAC directories.

- Copy the Hibernate files from the \hibernate-3.2 directory where you downloaded and unzipped the Hibernate files to the \DAC directory on the DAC Client machine, as described in the table below.

**Table 4–3** *Hibernate Library Files You Need to Copy*

Files	Copy from	Copy to
*.jar	\hibernate-3.2\lib	\DAC\lib
hibernate3.jar	\hibernate-3.2	\DAC\lib
hibernate-configuration-3.0.dtd	\hibernate-3.2\src\org\hibernate	\DAC
hibernate-mapping-3.0.dtd	\hibernate-3.2\src\org\hibernate	\DAC

## 4.9.2 Installing JDBC Drivers for DAC Database Connectivity

You must install the appropriate JDBC driver in the DAC\lib directory on the machines where the DAC Client is installed to enable DAC database connectivity.

### To install JDBC drivers in the \DAC\lib directory

- **Oracle.** If you are using an Oracle database (other than 8.x), find the directory where Oracle is installed. Copy the file named ojdbc14.jar in the jdbc\lib directory and paste it in the \DAC\lib directory.
  - If you are using Oracle 8.x, copy the file named classes12.zip and paste it in the OracleBI\DAC\lib directory. Also, edit the ORACLELIB setting in the config.bat file as follows:
 

```
set ORACLELIB=%DAC_HOME%\lib\classes12.zip
```
- **DB2.** If you are using a DB2 database, find the directory where DB2 is installed. In the Java subdirectory copy the file named db2java.zip and paste it in the \DAC\lib directory.

---

**Note:** If your source or target database is DB2-UDB, you also need to create DB2-UDB stored procedures to be used during the ETL process. For instructions, see [Section 4.18.2, "Creating Stored Procedures for DB2-UDB"](#).

---

- **MSSQL.** DAC is configured for Microsoft SQL Server 2005 JDBC drivers. If you are using a Microsoft SQL Server database, then download the Microsoft SQL Server 2005 JDBC Driver file sqljdbc.jar and copy it to the \DAC\lib\ directory.

You can use the Microsoft SQL Server 2000 JDBC Driver files if you edit the \conf\connection\_templates.xml file and un-comment the section that starts <!-- THIS SECTION IS FOR SQL SERVER 2000. Comment this section while using SQL Server 2005.-->.

Download the SQL Server JDBC drivers for SQL Server 2000 from the Microsoft web site. Copy the files msbase.jar, mssqlserver.jar, and msutil.jar to the \DAC\lib folder.

---



---

**Note:** You need the Microsoft SQL Server 2005 JDBC Driver 1.1 for SQL Server 2000 or SQL Server 2005.

---



---

- **Teradata.** If you are using a Teradata database, copy the files `tdgssconfig.jar`, `TdgssUserConfigFile.xml`, `terajdbc4.jar`, `log4j.jar`, and `tdgssjava.jar` from the Teradata installation directory to the `\DAC\lib` directory. Depending on the Teradata JDBC version, you might not have some of the above files.

### 4.9.3 Configuring the DAC Config.bat File

Follow these instructions to configure the DAC `config.bat` file.

#### To configure the DAC `config.bat` file

1. Navigate to the `\DAC` folder.
2. Open the `config.bat` file.
3. Edit the `JAVA_HOME` variable to point to the directory where you installed the Java SDK.

Make sure there are no spaces in the path reference.

For example:

```
set JAVA_HOME=d:\jdk1.5
```

4. Edit the `DAC_HOME` variable to point to the directory where you installed DAC.

For example:

```
set DAC_HOME=d:\OracleBI\DAC
```

### 4.9.4 Enabling DAC Client Communication with Informatica PowerCenter

The DAC Client uses the Informatica `pmrep` and `pmcmd` command line programs when communicating with Informatica PowerCenter. The DAC Client uses `pmrep` to synchronize DAC tasks with Informatica workflows and to keep the DAC task source and target tables information up to date.

This section includes the following topics:

- [Section 4.9.4.1, "How to Install Informatica `pmcmd` and `pmrep` Command Line Programs"](#)
- [Section 4.9.4.2, "How to Define the Informatica Domains File Path in the DAC Client Environment Variables"](#)
- [Section 4.9.4.3, "How to Verify the DAC Client Is Able to Use `pmrep` and `pmcmd`"](#)

#### 4.9.4.1 How to Install Informatica `pmcmd` and `pmrep` Command Line Programs

The `pmrep` program is installed in the Informatica PowerCenter Client and Informatica PowerCenter Services bin directories. Because of the requirement to co-locate the DAC Client with the PowerCenter Client, the `pmrep` program is available on the machine for the DAC Client to use.

The `pmcmd` program is installed in the PowerCenter Services bin directory. In order for the DAC to access `pmcmd`, it must also reside in the PowerCenter Client 8.1.1 SP5 bin directory on the same machine where the DAC Client is installed.

### To install pmcmd in the PowerCenter Client 8.1.1 SP5 bin directory on the DAC Client machine

Do one of the following:

- If PowerCenter Services 8.1.1 SP5 has been installed on the same machine as the DAC Client and PowerCenter Client 8.1.1 SP5, copy the pmcmd program from the PowerCenter 8.1.1\server\bin directory to PowerCenter 8.1.1\client\bin directory.
- If Informatica PowerCenter Services 8.1.1 SP5 is installed on a Windows machine other than the one that hosts the DAC Client, copy the pmcmd.exe file from the \PowerCenter 8.1.1\server\bin directory on the machine where PowerCenter Services is installed to the \PowerCenter 8.1.1\client\bin directory on the DAC Client machine.
- If Informatica PowerCenter Services is installed on a UNIX or Linux machine, and an installation on Windows is not available, run the Informatica PowerCenter Services installer on the DAC Client machine or (any other Windows machine) to install only Integration Services as follows:
  - a. Run the Informatica PowerCenter Services installer as described in section [Section 4.5.2, "Installing Informatica PowerCenter Services"](#).
  - b. In the Domain Configuration section, on the Create or Join Domain page, select the **Create New Domain** radio button
  - c. At the Configure Domain Database page, specify the database details for the domain configuration database.  
  
You may use the same database that stores the domain configuration for your PowerCenter Services installation on UNIX. However, you will have to create a different database account.
  - d. At the Configure Domain page, specify the domain details.  
**Note:** Do not check the Create Repository Service check box.
  - e. Complete the installation.
  - f. Apply Service Pack 4.
  - g. Copy the pmcmd.exe file located in PowerCenter 8.1.1\server\bin to PowerCenter 8.1.1\client\bin.

#### 4.9.4.2 How to Define the Informatica Domains File Path in the DAC Client Environment Variables

In order for the DAC Client to be able to use the pmrep and pmcmd programs, you need to define the path of the Informatica Domain file 'domains.infa' in the environment variables on the DAC Client machine.

##### To define the Informatica Domains File path in the DAC Client environment variables

1. Locate the file domains.infa in the root Informatica PowerCenter installation directory and note down the directory path of this file.  
  
For example, D:\Informatica\PowerCenter8.1.1.
2. Create an environment variable called INFA\_DOMAINS\_FILE with the value set to the directory path to the domans.infa file, as follows:
  - On Windows, display the Windows Environment Variables dialog box (that is, from the Windows Control Panel, select System, then Advanced, then

Environment Variables), and create a System variable with the following values:

- Variable name: INFA\_DOMAINS\_FILE
- Variable value: <directory path of domains file>\domains.infa

The path should include the name of the file. For example, 'D:\Informatica\PowerCenter8.1.1\domains.infa'.

3. Add the directory path to Informatica PowerCenter binaries to the PATH environment variable as follows:

In the Windows System Properties > Environment Variables dialog box, add the path of the Informatica \Client\bin directory to the PATH environment variable. For example: <drive>:\Informatica\PowerCenter8.1.1\client\bin.

#### 4.9.4.3 How to Verify the DAC Client Is Able to Use pmrep and pmcmd

From a Windows command prompt, execute pmrep and then pmcmd. The test is successful if you see the programs are invoked and the pmrep and pmcmd prompts appear.

If pmrep and pmcmd are not recognized, then:

- Ensure Service Pack 4 has been applied to Informatica PowerCenter 8.1.1.
- Verify that the INFA\_DOMAINS\_FILE variable points to the domains.infa file located in the Informatica directory.
- Verify that the PATH variable includes the path to the Informatica binaries (\Informatica\PowerCenter\client\bin), and that pmrep.exe and pmcmd.exe exist in the \bin folder.

## 4.10 Logging into DAC for the First Time and Creating the DAC Repository

When you log into DAC for the first time, you create a set of stored login information. Also, as part of the first-time login process, DAC prompts you to create the DAC Repository schema. Once the schema is created, you can then import metadata into the DAC Repository.

This section contains the following topics:

- [Section 4.10.1, "Logging into DAC"](#)
- [Section 4.10.2, "Creating the DAC Repository Schema"](#)
- [Section 4.10.3, "Importing Metadata into the DAC Repository"](#)

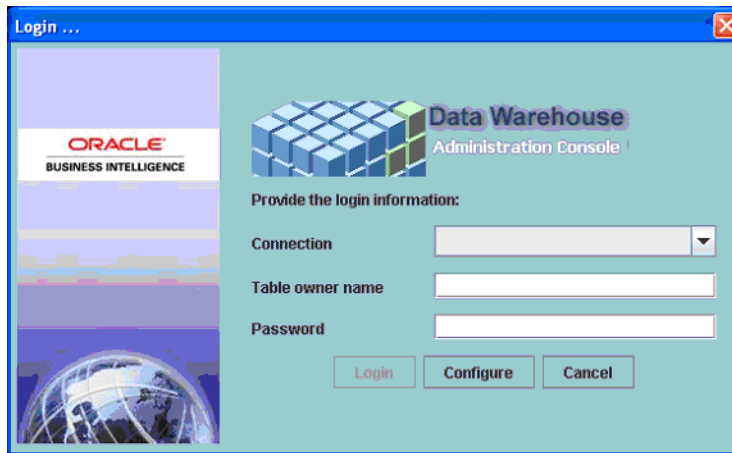
### 4.10.1 Logging into DAC

This section provides instructions for logging into DAC by creating a set of stored information.

**Note:** Before you start this procedure, you need to have already created a database for the DAC Repository. For more information, see [Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components."](#)

#### To log into DAC

1. Start the DAC Client by navigating to the \DAC directory and double-clicking the startclient.bat file.




---

**Note:** If you installed the DAC Client by running the Oracle BI Applications installer, you can also start the DAC Client from the Start menu > Oracle Business Intelligence > Oracle DAC > DAC Client.

---

2. In the Login... dialog box, select Configure.
3. In the Configuring... dialog box, select Create Connection, and then click Next.
4. Enter the appropriate connection details as specified in the table below.

Field	Required Value
Name	Enter a unique name for the connection to the DAC Repository.
Connection type	Select the type of database in which the DAC Repository is stored.
Connection String, or Database name, or TNS Name, or Instance	Specify the database name or database account name of the DAC Repository. If you are using: <ul style="list-style-type: none"> <li>■ Oracle (OCI8), use the tnsnames entry.</li> <li>■ Oracle (Thin), use the instance name.</li> <li>■ SQL Server, use the database name.</li> <li>■ DB2-UDB, use the connect string as defined in the DB2 configuration.</li> </ul>
Database Host	Enter the machine name of the machine where the DAC Repository database resides.
Database Port	Enter the port number on which the database listens. For example, for an Oracle database the default port is 1521, or for a SQL Server database the default port is 1433.

5. To test the connection, select Test Connection and enter the Table owner name and password for the DAC Repository.
6. Select Apply to save the connection details and return to the Login... dialog box.
7. Select the new connection from the Connection drop-down list, specify a table owner name and password for the DAC Repository, and click Login.

The DAC Repository that you connect to does not contain the DAC Repository tables. In the following section ([Section 4.10.2, "Creating the DAC Repository Schema"](#)), you will create the DAC Repository schema.

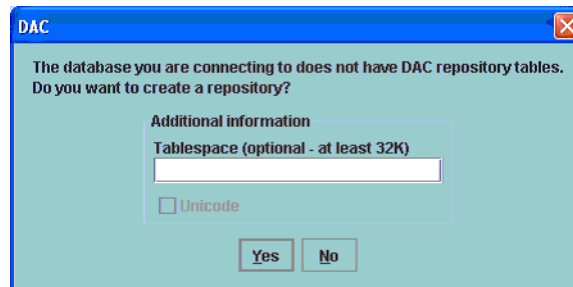
## 4.10.2 Creating the DAC Repository Schema

Follow these instructions to create the DAC Repository schema.

When you log into DAC and connect to a DAC Repository for the first time, DAC detects that the DAC schema does not exist in the database, and you are asked whether you want to create a repository.

### To create the DAC Repository schema

- After logging into DAC for the first time, when you are asked whether you want to create a repository, click Yes.



For a repository on Oracle or DB2 database, you have the option to specify a tablespace.

The Unicode check box is available for a repository on MS SQL Server or DB2 database. Check the Unicode check box if your deployment requires a Unicode schema to be created. To determine if your deployment requires a Unicode schema for the DAC Repository, see [Section 4.3.3, "Code Page and Data Movement Requirements"](#).

## 4.10.3 Importing Metadata into the DAC Repository

This section explains how to import metadata into the DAC Repository.

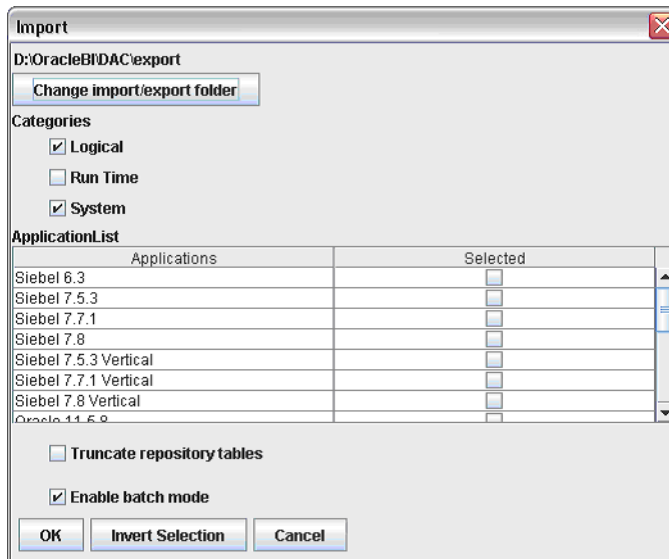
---

**Note:** The DAC Client can connect to only one DAC Repository at a time.

---

### To import metadata into the DAC Repository

- In DAC, select Tools, then DAC Repository Management, then Import to display the Import dialog box.



Make sure that the correct \DAC\export\ directory is displayed at the top of the dialog box (for example, <drive letter>:\OracleBI\DAC\export). If necessary, use the 'Change import/export directory' button to select the \DAC\export\ directory. If the wrong directory is selected, the Applications list will be empty

2. Select the Logical check box, and the System check box.
3. In the Applications List table, use the check boxes in the Selected column to specify the source system applications for which you will import the ETL metadata, as follows:
  - a. Select the check box next to Universal.  
To avoid possible ETL errors in other applications (for example, missing tasks), you must import the Universal application.
  - b. Select the check box for any other business applications that you want to deploy.

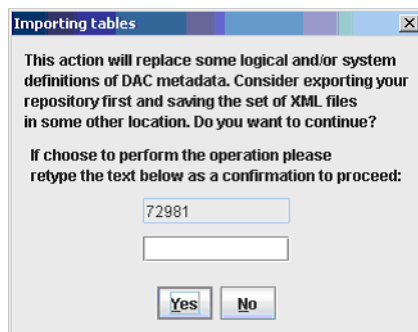
**Tip:** Make a note of the Application names that you select. When you create data warehouse tables later in the configuration process, you might need to type in the names exactly as they are displayed here (for more information, see [Section 4.11.1, "Creating Data Warehouse Tables"](#)).

4. Select the Categories options as follows:

Categories Options	Description
Logical	Imports all information contained in the DAC Design view and the execution plan information for the DAC Execute view.
Run Time	Imports ETL Run History and the last refresh date information.
System	Imports all information contained in the DAC Setup view, except passwords for servers and database connections.

5. Click OK to display the Importing tables dialog box.





6. To confirm that you want to import the seed data selected, re-type the text in the text box and click Yes.
7. If prompted to verify that you want to continue with the Import, click Yes.  
When the process is complete, the DAC displays a status box containing a success or failure message. If the process fails, use the \DAC\log\import.log file to diagnose errors.

---

**Note:** The import process typically takes between one and five hours, depending on the location of the database that stores the DAC Repository, the speed of the network, and the number of applications selected in the **Applications** list on the Import dialog box.

---

8. Click OK.
9. If your source or target database is a DB2/390 database, run DAC\_DB2390\_Source.sql or DAC\_DB2390\_Target.sql immediately after importing the seed data by connecting to the database where the DAC Repository resides. These files are stored in the \DAC\ directory.

## 4.11 Creating the Oracle Business Analytics Warehouse Tables

This section explains how to create tables in the Oracle Business Analytics Warehouse database.

**Note:** Before you start this procedure, you need to create a database for the Oracle Business Analytics Warehouse. For more information, see [Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components."](#)

The Oracle Business Analytics Warehouse tables are created by the DAC Client. The DAC Client uses ODBC connections to the Oracle Business Analytics Warehouse database for this procedure. Ensure that you have created an ODBC connection to the Oracle Business Analytics Warehouse database as described in [Section 4.3.4.3, "Create ODBC Database Connections."](#)

If your Oracle Business Analytics Warehouse database is Oracle, you must use the Oracle Merant ODBC driver to create the ODBC connection. This driver is installed by the Oracle BI Applications. The DAC Client must be installed by running the Oracle BI Applications installer so that the driver is installed on the machine. Create the ODBC DSN to the Oracle Business Analytics Warehouse as described in [Section 4.3.4.3, "Create ODBC Database Connections."](#)

Before you perform the procedures described in this section, make sure that the SSE role has been created for the Oracle Business Analytics Warehouse, and that the

database user has been associated with the role (for more information, see [Section 4.3.4.1.1, "How to Create the SSE Role"](#)).

If your Oracle Business Analytics Warehouse database is not Teradata, refer to [Section 4.11.1, "Creating Data Warehouse Tables"](#) to create the data warehouse tables.

If you are using a Teradata database as the Oracle Business Analytics database, the DAC creates a SQL file to create the schema tables, not the tables themselves. Follow the steps in [Section 4.11.2, "Creating Data Warehouse Tables on a Teradata Database"](#).

**Note:** Additional work by the database administrator is required if the data warehouse tables need to be moved to different tablespaces for performance or manageability reasons.

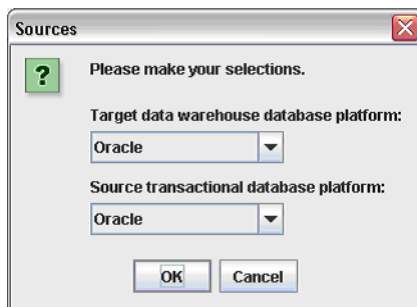
### 4.11.1 Creating Data Warehouse Tables

Use this procedure to create the data warehouse tables.

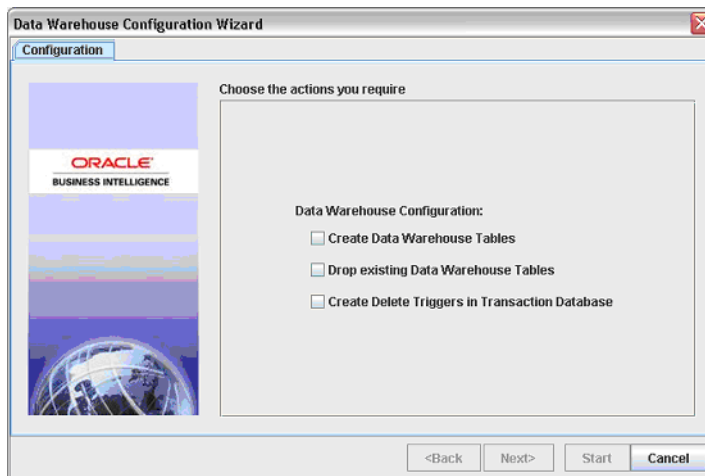
If the Oracle Business Analytics Warehouse database is Teradata, follow the steps in [Section 4.11.2, "Creating Data Warehouse Tables on a Teradata Database."](#)

#### To create data warehouse tables

1. In DAC, select Tools, then ETL Management, then Configure.  
For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)
2. In the Sources dialog box, select the database platform for the target data warehouse and source transactional database.

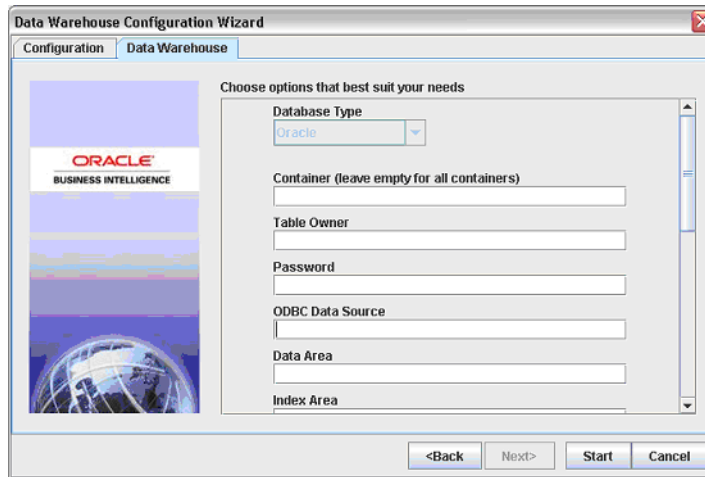


3. Click OK to display the Data Warehouse Configuration Wizard.



4. Select the Create Data Warehouse Tables check box, and click Next.

The Data Warehouse tab is active.



5. Enter the details of the database in which you want to store the data warehouse.

The information that you need to enter is dependent on the type of target database that you are using to store the data warehouse.

Field	Description
Database Type	(Read only) Type of database, as specified by the 'Target data warehouse database platform' field on the Sources dialog box (see step 2). If the database type is DB2/390, then check the 390 Database check box.

Field	Description
Container	<p>The name of the source business applications for which you want to create the data warehouse tables.</p> <p><b>Note:</b> You must leave this field blank, unless told to specify a value by Oracle Support.</p> <p>If you leave the Container field blank, DAC creates a container by default for all of the following:</p> <ul style="list-style-type: none"> <li>■ The source business applications that you selected when you imported the seed data into the DAC metadata repository earlier (for more information, see <a href="#">Section 4.10.3, "Importing Metadata into the DAC Repository"</a>).</li> <li>■ Any copies of those source system applications.</li> <li>■ Any additions made those source system applications.</li> <li>■ Any new source system applications that you have custom built.</li> </ul> <p>For example, you might do the following:</p> <ul style="list-style-type: none"> <li>■ Import the seed data for Oracle 11.5.9 and 11.5.10</li> <li>■ Make a copy of Oracle 11.5.10 and call it 'My Oracle 11.5.10'.</li> <li>■ Add two data warehouse tables to 'My Oracle 11.5.10'</li> <li>■ Create a new source system application called 'My new source' and add 10 data warehouse tables to 'my New Source'.</li> </ul> <p>If you leave the Container field blank, DAC will create the following containers:</p> <ul style="list-style-type: none"> <li>■ Oracle 11.5.9</li> <li>■ Oracle 11.5.10</li> <li>■ My Oracle 11.5.10</li> <li>■ My new Source</li> </ul> <p>If there are tables that are common to these containers, then only one table will be created. For example, if there is a table called W_ORG_D in Oracle 11.5.9 and Oracle 11.5.10, then DAC will create only one table called W_ORG_D. If columns are different for the same table across containers, then DAC will create a table that has all the columns in the same table.</p> <p>If you only want to deploy a subset of the source business applications for which you imported seed data earlier, then use this field to specify a container name. When you specify a container name, you must enter the names of the applications exactly as they are displayed on the seed data Import dialog box.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>■ If you want to deploy Oracle 11.5.9, enter: 'Oracle 11.5.9'.</li> <li>■ If you want to deploy Oracle 11.5.9 and 11.5.10, enter: 'Oracle 11.5.9,Oracle 11.5.10'.</li> </ul> <p>For more information about containers, see <i>Oracle Business Intelligence Data Warehouse Administration Console Guide</i>.</p>
Table Owner	<p>Valid database owner, username, or account that you set up to hold the data warehouse (for more information, see <a href="#">Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"</a>).</p>
Password	<p>Valid database user password for the database owner, username, or account that you specified in the Table Owner field (for more information, see <a href="#">Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"</a>).</p>

Field	Description
ODBC Data Source	Data Source Name (DSN) for the Oracle Business Analytics Warehouse. You must specify the name of the ODBC connection (for example, Connect_to_OLAP) that you created for the data warehouse. For more information, see <a href="#">Section 4.3.4.3, "Create ODBC Database Connections"</a> .
Data Area	(Optional) Tablespace where data warehouse tables are created.
Index Area	(Optional) Indexspace where data warehouse indexes are created (applicable only to Oracle and DB2 databases).
Is Unicode	Specifies whether the data warehouse database is Unicode. The database must be set to support the creation of a Unicode schema. For more information, see <a href="#">Section 4.3.3, "Code Page and Data Movement Requirements."</a>

6. Click Start.

The Run Status tab displays information about the process, as follows:

- If a 'Success' message is displayed, the data warehouse tables have been created. If you want to see log information about the process, use the following log files.
  - \OracleBI\DAC\config\generate\_ctl.log - A log of the schema definition process, including details of any conflicts between containers.
  - \OracleBI\DAC\config\createtables.log - A log of the ddlimp process.
- If a 'Failure' message is displayed, the data warehouse tables have not been created. Use the log information in \OracleBI\DAC\config\generate\_ctl.log to diagnose the error. The createtables.log is not generated.

### 4.11.2 Creating Data Warehouse Tables on a Teradata Database

This section explains how to create and delete data warehouse tables on a Teradata database. Before you start, make sure that the tables that you create are case specific by setting the session mode to ANSI in your Teradata ODBC configuration (for more information, see [Section 3.6, "Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse"](#)).

---

**Note:** For Unicode environments, perform these procedures using a login that has the default character set UTF-8.

---

#### To create data warehouse tables on a Teradata database

1. In DAC, select Tools, then ETL Management, then Configure.
 

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)
2. In the Sources dialog box, select 'Teradata' as the database platform for the source data warehouse, and select the appropriate database type for the target transactional database.
3. Click OK to display the Data Warehouse Configuration Wizard.
4. Click the 'Generate create statements for Data Warehouse Tables' check box, then click Next to display the Data Warehouse SQL tab.
5. Use the Container field to specify individual containers, or leave blank to deploy all containers.

6. Click Start.

The Run Status tab displays information about the process, as follows:

- If a 'Success' message is displayed, the data warehouse tables have been created. If you want to see log information about the process, use the following log files.
  - `\OracleBI\DAC\config\generate_ctl.log` - A log of the schema definition process, including details of any conflicts between containers.
  - `\OracleBI\DAC\config\createtables.log` - A log of the ddlimp process.
- If a 'Failure' message is displayed, the data warehouse tables have not been created. Use the log information in `\OracleBI\DAC\config\generate_ctl.log` to diagnose the error. The `createtables.log` is not generated.

7. Copy the SQL file created in step 6 from `\conf\sqlgen\sql\Teradata` into SQL Assistant and execute the SQL.

## 4.12 Installing the DAC Server

This section contains the following topics:

- [Section 4.12.1, "Installing an Additional Instance of the DAC Server on Windows"](#)
- [Section 4.12.2, "Installing the DAC Server on UNIX"](#)

### 4.12.1 Installing an Additional Instance of the DAC Server on Windows

The DAC Server is installed by the Oracle BI Applications installer on Windows. So, if you performed the steps in [Section 4.4, "Running the Oracle BI Applications Installer on Windows,"](#) you have already installed the DAC Server on the machine where you ran the Oracle BI Applications installer. This section provides instructions for installing an additional instance of the DAC Server on a machine different from the machine on which you ran the Oracle BI Applications installer.

**Note:** You do not need Oracle Business Intelligence Enterprise Edition installed on a machine to run the DAC Server. However, if you want to use the Oracle BI Applications installer to install the DAC Server on a Windows machine, you do need a full installation of Oracle Business Intelligence Enterprise Edition.

When installing a DAC Server, note the following prerequisites:

- Make sure that JDK 1.5 or higher is installed. For more information, see [Section 4.3.4.5, "Download and Install the Java JDK"](#).
- Make sure that Informatica PowerCenter Services is co-located on the machine that will host the DAC Server. For more information, see [Section 4.5, "Installing Informatica PowerCenter Services Software"](#).

The domain should be set properly and the server port set as a custom property.

- Make sure that database connectivity software has been installed for connection to source(s), to the Oracle Business Analytics Warehouse, and to the Informatica Repository/DAC Repository databases. For more information, see [Section 4.3.4.2, "Install Database Connectivity Software."](#)

#### To install an additional instance of the DAC Server

Do one of the following:

- If the target machine already has an Oracle Business Intelligence Enterprise Edition installed, you can run the Oracle BI Applications installer on the target machine to install the DAC Server. For information about how to run the Oracle BI Applications installer, see [Section 4.4, "Running the Oracle BI Applications Installer on Windows."](#)
- (Recommended) You can install a DAC Server by copying across the \DAC\ directory from the Windows machine on which you ran the Oracle BI Applications installer, as follows:

1. Copy the \OracleBI\DAC\ directory from the DAC Client machine to the target machine.

**Note:** The \DAC directory name on the target machine should not contain spaces.

---

**Note:** If you have copied the DAC folder from the DAC Client machine and the DAC Client has been configured to use the Hibernate libraries, you do not have to install the Hibernate libraries again. Otherwise, install Hibernate files following instructions in described in [Section 4.9.1, "Copying Hibernate Files to the DAC Directories."](#)

---

2. Copy the appropriate jdbc driver(s) to the \DAC\lib folder on the target machine. Follow instructions in described in [Section 4.9.2, "Installing JDBC Drivers for DAC Database Connectivity."](#)
3. Configure DAC config.bat file, as follows:
  - a. In the DAC directory, open the config.bat file.
  - b. Edit the JAVA\_HOME variable to point to the directory where you installed the Java SDK.

Make sure there are no spaces in the path reference.

For example:

```
set JAVA_HOME=d:\jdk1.5
```

4. Edit the DAC\_HOME variable to point to the directory where you installed the DAC.

For example:

```
set DAC_HOME=d:\OracleBI\DAC
```

## 4.12.2 Installing the DAC Server on UNIX

Oracle does not provide an installer for Oracle BI Applications on UNIX or Linux. Therefore, to install a DAC Server on UNIX or Linux, you copy the \DAC folder from the Windows machine on which the DAC Client is installed to a UNIX or Linux machine, as described below.

When installing a DAC Server, note the following prerequisites:

- Make sure that JDK 1.5 or higher is installed. For more information, see [Section 4.3.4.5, "Download and Install the Java JDK."](#)
- On UNIX or Linux, make sure that a zip and unzip utility is installed.

- Make sure that Informatica PowerCenter Services is co-located on the machine that will host the DAC Server. For more information, see [Section 4.5, "Installing Informatica PowerCenter Services Software."](#)

The domain should be set properly and the server port set as a custom property.

- Make sure that database connectivity software has been installed for connection to source(s), to the Oracle Business Analytics Warehouse, and to the Informatica Repository/DAC Repository databases. For more information, see [Section 4.3.4.2, "Install Database Connectivity Software."](#)

#### To copy over the DAC Server to a UNIX or Linux machine

1. On the Windows machine on which the DAC Client is installed, create a temporary directory (for example, a directory named \OracleBI\_UNIX\).

You will use this temporary directory to create a zip file for the UNIX or Linux deployment.

2. Copy the \DAC\ directory to the temporary directory (for example, \OracleBI\_UNIX\).
3. From the \DAC\ directory in the temporary directory, remove the \export and \icons subfolders.
4. Zip up the temporary directory (for example, \OracleBI\_Solaris\).
5. Copy the ZIP file to the target UNIX or Linux machine.

**Note:** If you use FTP to copy the zip file, use binary mode.

6. On the target UNIX or Linux machine, place the zip file in a directory where you want to install the DAC Server.
7. On the target machine, unzip the zip file.

Shell scripts are provided in the /DAC directory. After copying these files to a UNIX or Linux machine and before using them, you might need to use a MS-DOS to UNIX conversion tool to convert the script files to UNIX format (that is, remove the carriage return and line feed characters). Alternatively, you can manually remove the carriage return and line feed characters from the script files. For more information, see [Section A.7, "About the DAC Server Scripts for Bash and C Shells."](#)

8. On the UNIX or Linux machine copy the appropriate jdbc driver(s) to the /DAC/lib directory as described below. You must copy all jdbc drivers for the database platforms that host source(s), Oracle Business Analytics Warehouse, and DAC Repository databases:

- Oracle. If you are using an Oracle database (other than 8.x), locate the directory where the Oracle client is installed on the machine. Copy the file named ojdbc14.jar in the jdbc/lib directory and paste it in the \$DAC/lib folder, as described below.

**Note:** Do not copy the ojdbc14.jar file from a different machine.

If you are using Oracle 8.x, copy the file named classes12.zip and paste it in the \$DAC/lib folder. Also, edit the ORACLELIB setting in the config.sh or config.csh file as follows:

```
setenv ORACLELIB=$DAC_Home/lib/classes12.zip
```

The config.sh and config.csh files are located in the /DAC/ directory.



- DB2. If you are using a DB2 database, find the directory where the DB2 client is installed on the machine. In the Java sub-directory, copy the file named db2java.zip and paste it in the \$DAC/lib folder.

**Note:** Do not copy the db2java.zip file from a different machine.

- MSSQL. DAC is configured for Microsoft SQL Server 2005 JDBC drivers. If you are using a Microsoft SQL Server database, then download the Microsoft SQL Server 2005 JDBC Driver file sqljdbc.jar and copy it to the /DAC/lib directory.

**Note:** You need the Microsoft SQL Server 2005 JDBC Driver 1.1 for SQL Server 2000 or SQL Server 2005.

You can use the Microsoft SQL Server 2000 JDBC Driver files if you edit the /DAC/conf/connection\_templates.xml file and un-comment the section that starts <!-- THIS SECTION IS FOR SQL SERVER 2000. Comment this section while using SQL Server 2005.--> Download the SQL Server JDBC drivers for SQL Server 2000 from the Microsoft web site. Copy the files msbase.jar, mssqlserver.jar, and msutil.jar to the /DAC/lib folder.

- Teradata. If you are using a Teradata database, copy the files tdgssconfig.jar, TdgssUserConfigFile.xml, terajdbc4.jar, log4j.jar, and tdgssjava.jar from the Teradata installation directory to the /DAC/lib directory. Depending on the Teradata JDBC version, you might not have some of the above files.

9. If required, edit the config.sh or config.csh file located in the /DAC directory to point to the correct version of the Java JDK by setting the JAVA\_HOME environment variable.

## 4.13 Configuring the DAC Server

This section contains instruction for configuring the DAC Server. It contains the following topics:

- [Section 4.13.1, "Configuring the Connection Between the DAC Server and DAC Repository"](#)
- [Section 4.13.2, "Enabling DAC Server Communication with Informatica PowerCenter"](#)
- [Section 4.13.3, "Starting the DAC Server"](#)
- [Section 4.13.4, "Activating Join Indexes for Teradata Databases"](#)

### 4.13.1 Configuring the Connection Between the DAC Server and DAC Repository

You must configure the connection between the DAC Server and the DAC Repository.

On Windows, you can use the DAC Client to configure a DAC Server that runs in the same \DAC folder. Optionally, or to configure a DAC Server installed in another folder or on another Windows machine, use the serverSetupPrompt.bat file to configure the repository connection.

On UNIX or Linux, use the serverSetupPrompt.sh or serverSetupPrompt.csh script to configure the connection between the DAC Server and the DAC Repository, as follows:

- To configure the DAC Server repository connection using the DAC Client, see [Section 4.13.1.1, "How to Configure the DAC Server Repository Connection Using the DAC Client \(Windows\)"](#).

- To configure the DAC Server repository connection using the serverSetupPrompt scripts, see [Section 4.13.1.2, "How to Configure the DAC Server Repository Connection Using serverSetupPrompt Scripts \(Windows, UNIX or Linux\)"](#).

### 4.13.1.1 How to Configure the DAC Server Repository Connection Using the DAC Client (Windows)

If the DAC Server is co-located with a configured DAC Client in the same \DAC folder, you can set the connection between the DAC Server and DAC Repository using the DAC Client, as described below.

#### To configure the connection between the DAC Server and the DAC Repository using the DAC Client

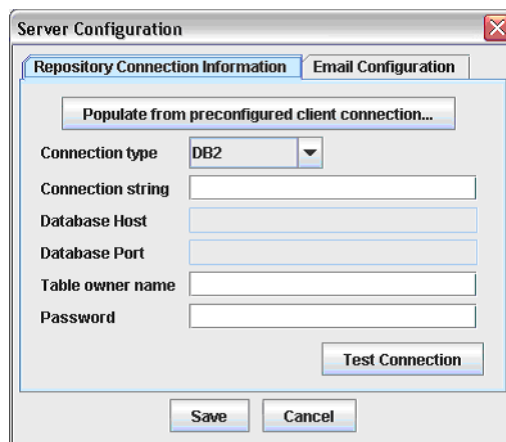
1. In DAC, select Tools, then DAC Server Management, then DAC Server Setup.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

**Note:** The DAC Repository that you connect to using the DAC Client is the one that will store the DAC Server repository connection information that you will specify in this procedure.

A confirmation dialog box asks you to confirm that you want to configure the DAC Repository connection.

2. Click Yes to display the Server Configuration dialog box.



3. In the Repository Connection Information tab, enter the appropriate information, as described in the table below.

**Tip:** If the DAC Server is running on the same machine as the DAC Client, click **Populate from preconfigured client connection** to populate the fields with connection details from the DAC Client.

Field	Description
Connection type	Select the type of database that you are using to store the DAC metadata repository. Depending on what type you select, the connection details below change (see <i>Connection fields</i> below).

Field	Description
Connection fields (for example, Instance, TNS Name, Connection string/Database name).	<p>Specify connection details for the database that stores the DAC metadata repository.</p> <ul style="list-style-type: none"> <li>■ If you select Oracle (Thin), you are prompted below for the following information: <ul style="list-style-type: none"> <li>■ Instance (for example, mymachinename).</li> <li>■ Database Host (fully qualified, for example, mymachine.us.company.com).</li> <li>■ Database Port (for example, 1521).</li> <li>■ Table owner name, using the account that you created a database to store the DAC Repository (for example, DAC). For more information, see <a href="#">Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"</a>.</li> <li>■ Password (that is, for the above database account).</li> </ul> </li> <li>■ If you select Oracle (OCI8), you are prompted below for a TNS name (for example, mymachinename@host.com).</li> <li>■ If you select DB2, you are prompted below for a Connection string.</li> <li>■ If you select MS SQL Server, you are prompted below for a Database name (for example, mydacdatabase).</li> </ul>
Table owner name	The database table name or instance (for example, DAC) that you created to store the DAC Repository (for more information, see <a href="#">Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"</a> ).
Password	The database or instance password (for example, DAC).

---

**Note:** The DAC Repository details that you specify here must match the DAC Repository details that you specified in the following tasks:

- When you created a database to store the DAC Repository (for more information, see [Section 4.3.4.1, "Create Databases for Oracle BI Applications and Informatica PowerCenter Components"](#)).

- When you created a DAC connection (for more information, see [Section 4.10.1, "Logging into DAC"](#)).

---

4. Click Test Connection to make sure the DAC Repository connection works.
5. Click Save.

### 4.13.1.2 How to Configure the DAC Server Repository Connection Using serverSetupPrompt Scripts (Windows, UNIX or Linux)

Use the serverSetupPrompt.sh or serverSetupPrompt.csh file to configure the connection between the DAC Server and the DAC Repository when the DAC Server is installed on UNIX or Linux, as described below. For more information on the DAC Server scripts, see [Section A.7, "About the DAC Server Scripts for Bash and C Shells."](#)

Use the serverSetupPrompt.bat file to configure the connection between the DAC Server and the DAC Repository when the DAC Server is installed on Windows, as described below.

To configure the connection between the DAC Server and the DAC Repository using the serverSetupPrompt files:

1. Run the serverSetupPrompt script, as follows:
  - On Windows, double-click the serverSetupPrompt.bat located in the \DAC directory.
  - On UNIX or Linux, run serverSetupPrompt.sh or serverSetupPrompt.csh located in the /DAC directory.
2. Enter 1 in the 'Please make your selection' prompt to enter repository connection information.
3. Enter the number for the type of database storing the DAC Repository from the list of connection type choices.
4. Enter the connection information as described in the Connection Information table in section [Section 4.13.1.1, "How to Configure the DAC Server Repository Connection Using the DAC Client \(Windows\)"](#).
5. Enter 2 to test the DAC Repository connection.
6. Enter 5 to save changes.
7. Enter 6 to exit.

## 4.13.2 Enabling DAC Server Communication with Informatica PowerCenter

The DAC Server uses the following command line programs to communicate with Informatica PowerCenter:

- pmrep is used to communicate with PowerCenter Repository Services.
- pmcmd is used to communicate with PowerCenter Integration Services.

The DAC Server uses the pmcmd program to run the Informatica workflows using PowerCenter Integration Services. The pmrep and pmcmd programs are installed during the PowerCenter Services installation in the bin folder of the server directory. Oracle recommends that you co-locate the PowerCenter Services with the DAC Server so that the pmrep and pmcmd programs are available on the machine for the DAC Server to use. To enable the DAC Server to communicate with Informatica PowerCenter Services using the pmrep and pmcmd programs, you need to define the path of the Informatica Domain file 'domains.infa', and set appropriate environment variables as follows:

- If your DAC Server is installed on Windows, see [Section 4.13.2.1, "How to Set Environment Variables on Windows"](#).
- If your DAC Server is installed on UNIX or Linux, see [Section 4.13.2.2, "How to Set Environment Variables on UNIX or Linux"](#).

### 4.13.2.1 How to Set Environment Variables on Windows

To set the environment variables on Windows:

1. Locate the file `domains.infa` in the root Informatica PowerCenter installation directory and note down the directory path of this file.

For example, `D:\Informatica\PowerCenter8.1.1`.

2. Create an environment variable called `INFA_DOMAINS_FILE` with the value set to the directory path to the `domains.infa` file, as follows:
  - On Windows, display the Windows Environment Variables dialog box (that is, from the Windows Control Panel, select System, then Advanced, then Environment Variables), and create a System variable with the following values:
    - Variable name: `INFA_DOMAINS_FILE`
    - Variable value: `<<directory path of domain file>\domains.infa`

The Variable value should include the `domains.infa` file name. For example, `D:\Informatica\PowerCenter8.1.1\domains.infa`.

3. Add the directory path to Informatica PowerCenter binaries to the `PATH` environment variable as follows:

In the Windows System Properties > Environment Variables dialog box, add the path of the `Informatica \server\bin` directory to the `PATH` environment variable. For example: `<drive>:\Informatica\PowerCenter8.1.1\server\bin`.

#### How to Verify the DAC Server Is Able to Use `pmrep` and `pmcmd`

From a Windows command prompt, execute `pmrep` and then `pmcmd`. The test is successful if the `pmrep` and `pmcmd` prompts appear.

If `pmrep` and `pmcmd` are not recognized, then:

- Ensure Service Pack 4 has been applied to Informatica PowerCenter 8.1.1.
- Verify that the `INFA_DOMAINS_FILE` variable points to the `domains.infa` file located in the Informatica directory.
- Verify that the `PATH` variable includes the path to the Informatica binaries (`\Informatica\PowerCenter\server\bin`).

### 4.13.2.2 How to Set Environment Variables on UNIX or Linux

Use the `dac_env.sh` or `dac_env.csh` file to set the appropriate environment variables on a UNIX or Linux machine. For more information on the DAC Server scripts, see [Section A.7, "About the DAC Server Scripts for Bash and C Shells."](#)

To set environment variables on a UNIX or Linux machine:

1. Open the `dac_env.sh` file or `dac_env.csh` file for editing.  
The files are located in the `/DAC` directory.
2. Locate the OS section for the operating system on which your DAC Server is installed.
3. Uncomment out the line to set the library path environment variable, and replace `%DAC_PMCMD_PATH%` with the path to the PowerCenter Services binaries.

For example, using the `dac_env.sh` file, if the DAC Server is on Solaris or Linux, in the `SOLARIS\LINUX` section, change the following line from:

```
#export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:%DAC_PMCMD_PATH%
```

To:

```
export LD_LIBRARY_PATH=${LD_LIBRARY_
PATH}:/home/Informatica/PowerCenter8.1.1/server/bin
```

4. In the Common section for Informatica Configuration, set the environment variables for PATH, PM\_HOME, INFA\_DOMAINS\_FILE as follows:
  - a. Replace %INFORMATICA\_SERVER\_LOCATION% with the path to Informatica PowerCenter Services installation.
  - b. Replace %DAC\_PMCMD\_PATH% with the path to Informatica PowerCenter Services binaries.
  - c. Replace %DOMAINS.INFA\_FILE\_LOCATION% with the path to the domains.infa file.

Include the name of the file. For example,  
/home/Informatica/PowerCenter8.1.1/domains.infa.

For example, using the dac\_env.sh file, change the following lines in the common section from:

```
export PATH=${PATH}:%DAC_PMCMD_PATH%:%INFORMATICA_SERVER_LOCATION%:.
export PM_HOME=%INFORMATICA_SERVER_LOCATION%
export INFA_DOMAINS_FILE=%DOMAINS.INFA_FILE_LOCATION%
```

To:

```
export PATH=${PATH}:/home/Informatica/PowerCenter8.1.1/server/bin
export PM_HOME=/home/Informatica/PowerCenter8.1.1/server
export PM_HOME=/home/Informatica/PowerCenter8.1.1/server
export INFA_DOMAINS_FILE= /home/Informatica/PowerCenter8.1.1/domains.infa
#export LANG=C (uncomment if required)
#export LC_ALL=C (uncomment if required)
```

5. If required, edit the dac\_env.sh file and change the source command from:

```
source ./dac_env.sh
```

To:

```
./dac_env.sh
```

### How to Verify the DAC Server on UNIX or Linux Is Able to Use pmrep and pmcmd

Invoke config.sh or config.csh to set environment. For example, `./config.sh`. Then, invoke pmcmd.

Make sure that invoking pmcmd starts the pmcmd shell. If you get a 'command not found' error, then the location of the PowerCenter Services is not properly added to PATH in dac\_env.sh or dac\_env.csh. Review all environment variable settings to ensure they are correctly set.

Then, invoke pmrep, and make sure that invoking pmrep starts the pmrep shell. If you get a 'command not found' error, then the location of the PowerCenter Services is not properly added to PATH in dac\_env.sh or dac\_env.csh. Review all environment variable settings to ensure they are correctly set.

On some shells, export commands in dac\_env.sh and config.sh might not work correctly. In this case, try breaking the commands in two. For example, from:

```
export JAVA_HOME=/opt/java1.5
```

To:

```
JAVA_HOME=/opt/java1.5
export JAVA_HOME
```

### 4.13.2.3 How to Verify Java JDK Availability and Version

To verify that the DAC Server uses the correct Java JDK:

1. Invoke config.sh or config.csh to set environment.

For example:

```
. ./config.sh
```

2. Verify Java availability and version by typing the following command:

```
$JAVA -version
```

The Java version is returned.

If you receive a 'command not found' error message, or the Java version is lower than 1.5, then the JAVA\_HOME parameter in config.sh or config.csh is pointing to a non-existent or incorrect Java JDK location.

### 4.13.3 Starting the DAC Server

On Windows, start the DAC Server by double-clicking the \DAC\startserver.bat script.

On UNIX or Linux, start the DAC Server by executing either startserver.sh or startserver.csh.

For more information on starting and stopping the DAC Server on Windows, UNIX or Linux, see [Section A.3, "How to Start and Stop the DAC Server."](#)

### 4.13.4 Activating Join Indexes for Teradata Databases

For Teradata databases, the preconfigured tasks for creating and dropping join indexes are inactive.

**To activate join indexes for Teradata databases**

1. In DAC, create a new execution plan with the list of subject areas that you are interested for the ETL.
2. Query for all tasks whose name starts with 'Teradata Drop' and add them as preceding tasks.
3. Query for all tasks whose name start with 'Teradata Create' and add them as following tasks.
4. Assemble the execution plan parameters in the Parameters tab and configure the parameters.
5. Redesign the execution plans.

## 4.14 Configuring DAC Integration Settings

This section explains how to configure DAC integration settings. It contains the following topics:

- [Section 4.14.1, "Setting DAC System Properties"](#)
- [Section 4.14.2, "Registering Informatica Services in DAC"](#)
- [Section 4.14.3, "Setting Physical Data Sources"](#)

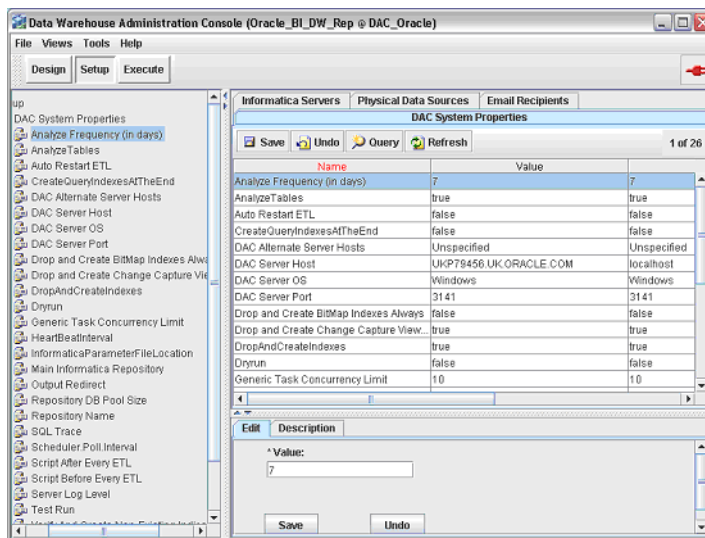
You must start the DAC Server to complete certain steps in the procedures described below. For information on how to start the DAC Server, see [Section A.3, "How to Start and Stop the DAC Server."](#)

### 4.14.1 Setting DAC System Properties

This section describes the DAC System Properties to set to ensure proper integration between the DAC Client, the DAC Server and Informatica.

#### To set DAC System Properties

1. Log in to DAC.  
For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)
2. From the Views menu, select Setup, then DAC System Properties tab.



3. Set values for the following properties:

Property	Value Required
DAC Server Host	Enter the name or IP address of the machine that hosts the DAC Server.
DAC Server OS	Enter the operating system of the machine that hosts the DAC Server. Possible values are Windows, AIX, Solaris, HP-UX, Linux (Case Sensitive).
InformaticaParameterFileLocation	Enter the path to the Informatica source file directory. For example, \Informatica\PowerCenter8.11\server\infa_shared\SrcFiles.



**Note:** For a list of DAC System Properties and their descriptions, see [Section A.2, "How to Set DAC System Properties."](#)

## 4.14.2 Registering Informatica Services in DAC

This section explains how to register the Informatica PowerCenter Integration Services service and the Informatica PowerCenter Repository Service in DAC.

When you register Informatica Services in DAC, note the following:

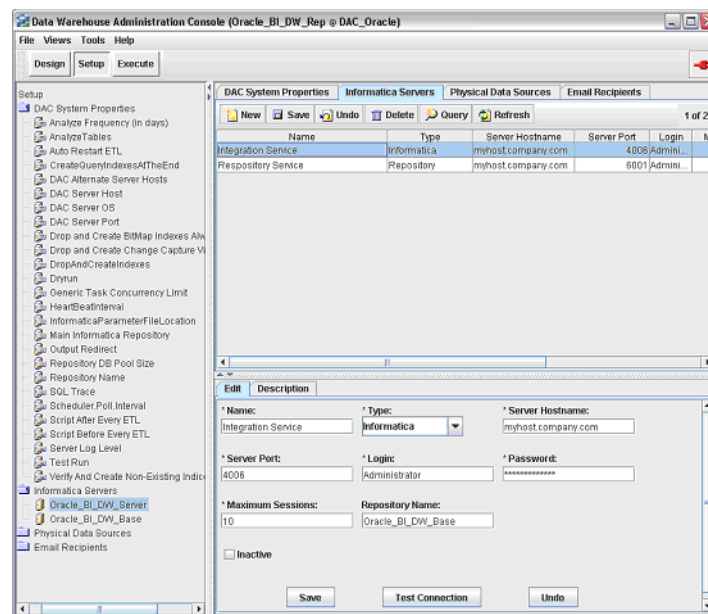
- You must register one or more Integration Services service in DAC.
- You must register one Repository Service in DAC.

**Tip:** For details of the PowerCenter Services that you need to register, log into the Informatica PowerCenter Administration Console. For more information, see [Section A.4, "How to Log Into Informatica PowerCenter Administration Console."](#)

### To register Informatica Services in DAC

1. In DAC, navigate to the Setup view, and then click the Informatica Servers tab.

For instructions on logging into DAC, see [Section A.1, "How to Log into DAC."](#)



2. From the Views menu, select Setup, then Informatica Servers.
3. For the Informatica Integration Service that you want to register, do the following:
  - a. Register the Informatica Integration Services service by modifying the record with Name = Oracle\_BI\_DW\_Server. To modify the record, enter the following information in the Edit subtab:

Field	Enter
Name	Logical name for the Integration Service (for example, Integration_Service).
Type	Select Informatica.

Field	Enter
Server Hostname	Integration Service host name or IP address. For example, myhost.company.com.
Server Port	The port that DAC uses to communicate with the Integration Services service (for example, 4006).  Specify the port number that you configured for the Integration Services service to listen on when setting the ServerPort custom property in <a href="#">Section 4.7.3, "Setting PowerCenter Integration Services Custom Properties"</a> .
Login	Informatica Repository user name who has appropriate privileges to execute workflows (for example, Administrator).
Password	Informatica Repository user password.
Maximum Sessions	Maximum number of workflows that can be executed in parallel on the Informatica PowerCenter Integration Services service. If the number of sessions is zero or is not specified, the DAC Server assigns the default value of 10.
Repository Name	Enter the name of the Oracle BI Applications repository that you restored in <a href="#">Section 4.6, "Restoring the Prebuilt Informatica Repository"</a> . For example, Oracle_BI_DW_Base (or Oracle_BI_DW_Teradata for Teradata installations).
Inactive	Indicates whether the Informatica PowerCenter Integration Services service is active or inactive.

- b. Click Test Connection to make sure that the connection works.

**Note:** Integration Services must be running (for more information about starting Integration Services, see [Section A.5, "How to Start and Stop Informatica Services."](#))

- c. Click Save to save the details.

4. For the Informatica Repository Service you want to register, do the following:

- a. Register the Informatica Repository Service by modifying the record with Name = Informatica\_REP\_Server. To modify the record, enter the following information in the Edit subtab:

Field	Enter
Name	Logical name for the Repository Service (for example, Repository_Service).
Type	Select Repository.
Server Hostname	Repository Service host name or IP address.
Server Port	Port where the Repository Service connects for requests. (The default port number is 6001.)
Login	Informatica Repository user name that has appropriate privileges to execute workflows (for example, Administrator).
Password	Informatica Repository user password (for example, Administrator).
Maximum Sessions	Maximum number of workflows that can be executed in parallel on the Informatica PowerCenter Integration Services service. If the number of sessions is zero or is not specified, the DAC Server assigns the default value of 10.

Field	Enter
Repository Name	Name of the Oracle BI Applications repository that you restored in <a href="#">Section 4.6, "Restoring the Prebuilt Informatica Repository."</a>
Inactive	Indicates whether the Informatica PowerCenter Integration Services service is active or inactive.

- b. Click Test Connection to make sure that the connection works.

**Note:** Integration Services must be running. For more information about starting Integration Services, see [Section A.5, "How to Start and Stop Informatica Services."](#)

- c. Click Save to save the details.

5. Click Test Connection to make sure that the connection works.
6. Click Save to save the details.

### 4.14.3 Setting Physical Data Sources

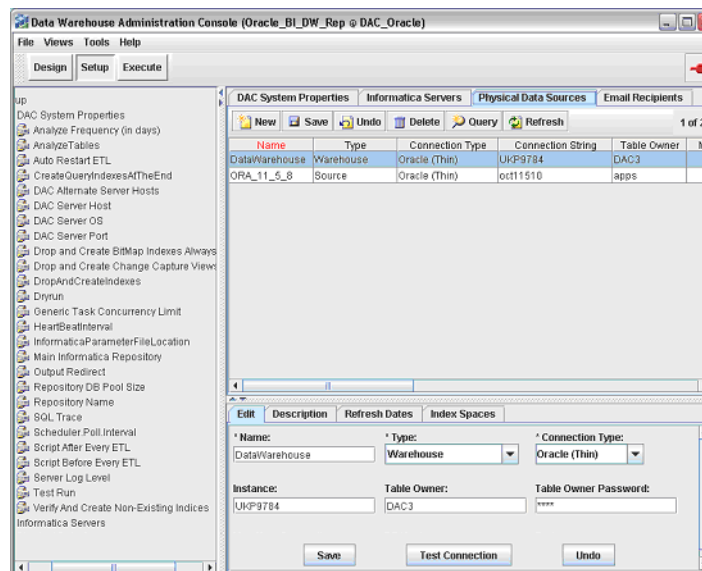
Follow this procedure to specify the transactional and data warehouse data sources in DAC.

#### To specify transactional and data warehouse data sources

1. Log into DAC.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

2. From the Views menu, select Setup, then Physical Data Sources.



The Physical Data Sources tab displays a precreated record for the data warehouse with name DataWarehouse, and one or more records for the OLTP sources. The records that are created by DAC for the OLTP sources depend on the business application source systems you selected when importing the DAC metadata. For more information, see [Section 4.10.3, "Importing Metadata into the DAC Repository."](#)

3. For each record, enter the following information in the Edit subtab:

Field	Description
Name	<p>Logical name for the OLAP or OLTP database connection. Do not change the default values. For example, you must specify DataWarehouse as the name of the OLAP data source.</p> <p><b>Note:</b> When you create an execution plan to perform an ETL, you need to use the OLAP and OLTP data source names that you specify here as the Parameter values for the execution plan parameters DBConnection_OLTP and DBConnection_OLAP. For more information about setting the Parameter values for an execution plan, see <i>Oracle Business Intelligence Data Warehouse Administration Console Guide</i>.</p>
Type	<ul style="list-style-type: none"> <li>■ 'Source' for the database connection for a transactional (OLTP) database.</li> <li>■ 'Warehouse' for the database connection for a data warehouse (OLAP) database.</li> </ul> <p>Do not change the default values.</p>
Connection Type	<p>Type of database. Possible values are:</p> <ul style="list-style-type: none"> <li>■ Oracle (OCI8)</li> <li>■ Oracle (Thin)</li> <li>■ DB2</li> <li>■ DB2/390</li> <li>■ MSSQL</li> <li>■ Teradata</li> <li>■ Flat File</li> </ul> <p>Select the Type based on your database.</p>
Instance or TNS Name or Connection String or Database Name (NOTE: The field name changes based on the Connection Type selection.)	<p>Enter the value appropriate for your database.</p> <p>For an Oracle database instance, enter the database SID (that is the SERVICE_NAME = value in the tnsnames.ora file in \network\admin\.</p> <p>For an Oracle TNS Name, enter the TNS entry name that is specified in the tnsnames.ora file in \network\admin\.</p>
Table Owner	Valid database user.
Table Owner Password	Valid database user password.
Max Num Connections	Maximum number of database connections this connection pool can contain.
DB Host	Machine name or instance where the database resides. For example, databasename.us.company.com.
Port	Port number where the database listens (for example 1521 is the default for an Oracle database).
Priority	Number used to generate dependencies when designing execution plans.

Field	Description
Data Source Number	<p>Unique number assigned to the data source category so that the data can be identified in the data warehouse. For example, the value '1' is used for Siebel data sources.</p> <p>If you are editing a data source template for a data source type, Oracle recommends that you do not change the default value. If you are specifying a data source without using a pre-defined template, you must use the correct value for that data source category. For example, if you specify an Oracle EBS R12 data source, you must specify the DATASOURCE_NUM_ID value '9'. For a complete list of supported data sources and DATASOURCE_NUM_ID values, see the section entitled, "How to Configure Data Source Num IDs," in <i>Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users</i>.</p> <p>This value is passed as a parameter to the Informatica workflows. If you are using multiple sources, each data source has a unique number. Typically, all source dependent extracts will use this parameter to populate the DATASOURCE_NUM_ID column, and the source independent workflows will carry these values to the final dimension and fact tables.</p>
Default Index Space	<p>(Oracle specific) Specifies the table space in which DAC drops and creates indexes against this database connection</p> <p>To use the default table space, leave this field blank.</p>

4. Click Test Connection to make sure the connection works.
5. Click Save.

## 4.15 Configuring Relational Connections in Informatica Workflow Manager

This section explains how to log into Informatica PowerCenter Workflow Manager and define relational connections. It contains the following topics:

- [Section 4.15.1, "Logging Into Informatica PowerCenter Workflow Manager"](#)
- [Section 4.15.2, "Configuring Relational Connections in Informatica PowerCenter Workflow Manager"](#)

### 4.15.1 Logging Into Informatica PowerCenter Workflow Manager

**Note:** Before you log into Workflow Manager, start the Informatica services. For more information, see [Section A.5, "How to Start and Stop Informatica Services."](#)

#### To log into Informatica Workflow Manager

1. On the Informatica PowerCenter Client machine, start Informatica PowerCenter Workflow Manager.
2. Choose Repository, then Add, to display the Add Repository dialog box.
3. In the Add Repository dialog box:
  - a. In the Repository field, specify Oracle\_BI\_DW\_Base (or the appropriate Oracle BI Applications repository name).
  - b. In the Username field, specify the Repository Administrator username (for example, Administrator).
4. Choose Repository, then Connect, to display the Connect to Repository dialog box.

5. In the Connect to Repository dialog box:
  - a. In the Password field, specify the Administrator password (for example, Administrator).
  - b. If the Connection Settings area is not displayed, click More.
  - c. Click Add (next to the **Domain** drop down list) to display the Add Domain dialog box.
6. In the Add Domain dialog box:
  - a. Specify the name of the domain that was created when you installed Informatica PowerCenter Services (for example, Domain\_<hostname>).
  - b. Specify the fully qualified hostname for the gateway host (for example, mymachine@us.company.com).
  - c. Specify the port for the gateway port (for example, 6001).
  - d. Click OK to save the details and close the Add Domain dialog box.
7. Click Connect.

## 4.15.2 Configuring Relational Connections in Informatica PowerCenter Workflow Manager

You need to use Informatica PowerCenter Workflow Manager to configure the relational connections, as follows:

- For all databases, follow the steps in [Section 4.15.2.1, "How to Configure Relational Connections."](#)
- Additionally, for Teradata databases, follow the steps in [Section 4.15.2.2, "\(Teradata specific\) How to Configure the Informatica Repository for Teradata External Loader Connections."](#)

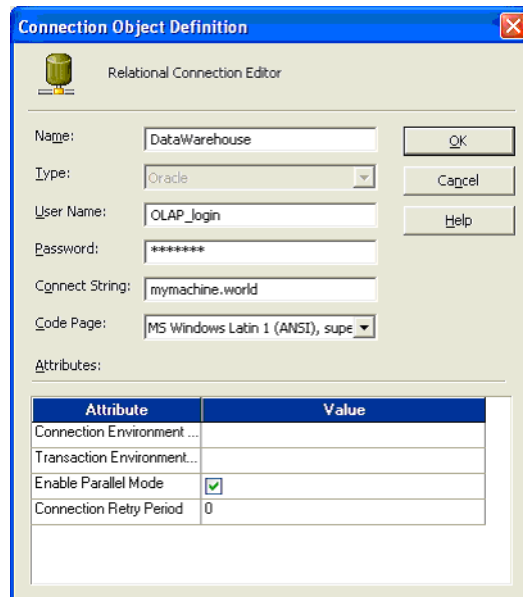
**Note:** The Informatica services must be running to perform these tasks. To start the Informatica services, see [Section A.5, "How to Start and Stop Informatica Services."](#)

### 4.15.2.1 How to Configure Relational Connections

Follow this procedure to configure relational connections:

1. In Informatica PowerCenter Workflow Manager, select Connections, then Relational to display the Relational Connection Browser.

You need to create a connection for each transactional (OLTP) database, and a connection for the Oracle Business Analytics Warehouse (OLAP) database.
2. For each database connection that you need to create, do the following:
  - a. Click New to display the Select Subtype dialog box, select the appropriate database type (for example, Oracle), then click OK.
  - b. Click New to display the Connection Object Definition dialog box, and enter the details.



- c. Click OK to save the details.

#### Notes:

- If the target database is Oracle, DB2, or Teradata, use the settings as follows:

- Click New and select the type as Oracle, DB2, then click OK.
- Name: DataWarehouse, for connection to the Oracle Business Analytics Warehouse.

You must specify 'DataWarehouse' exactly as it appears in the Physical Data Sources tab in the DAC Setup View. For the connection to the OLTP, specify the name exactly as it appears in the Physical Data Sources tab in the DAC Setup View. For example, if your source system is Siebel release 7.8.x, then you name this connection as SEBL\_78. For more information, see [Section 4.14.3, "Setting Physical Data Sources"](#).

- User Name: Database user name.
- Password: Database user password.
- Connect string: Connect string for the database (refer to the Informatica Help for information about specifying this value).
- Code Page: Code page compatible with the code page of the database client. If NLS\_LANG (for Oracle database) or DB2CODPAGE (for DB2 database) has been set, then the Code Page value should be a code page compatible with the language set by these variables. For more information, see [Section 4.3.3, "Code Page and Data Movement Requirements"](#).
- If the target database is SQL Server, use the settings as follows:
  - Click New and select the type as ODBC, then click OK.
  - Name: DataWarehouse, for the connection to the Oracle Business Analytics Warehouse. You must specify 'DataWarehouse' exactly as it appears in the Physical Data Sources tab in the DAC Setup View.
  - For the connection to the OLTP, specify the name exactly as it appears in the Physical Data Sources tab in the DAC Setup View. (For example, if

your source system is Siebel version 7.8.x then you name this connection as SEBL\_78). For more information, see [Section 4.14.3, "Setting Physical Data Sources"](#).

- User Name: Database user name.
- Password: Database password.
- Connect string: Connect string for the connecting to the database (refer to the Informatica Help for information about specifying this value).
- Code Page: Code page compatible with the code page of the database client.

3. Click Close to close the Relational Connection Browser.

#### 4.15.2.2 (Teradata specific) How to Configure the Informatica Repository for Teradata External Loader Connections

To configure the Informatica Repository for Teradata, you need to do the following:

- Specify Loader Connections for Teradata (for more information, see [Section 4.15.2.2.1, "How to Specify Loader Connections for Teradata"](#)).
- For each workflow, specify Loader Connection details at the session level (for more information, see [Section 4.15.2.2.2, "How to Specify Teradata Details at the Workflow Level"](#)).

**4.15.2.2.1 How to Specify Loader Connections for Teradata** Follow this procedure to configure the Informatica Repository in Informatica PowerCenter Workflow Manager for Teradata external loader connections.

1. In Informatica PowerCenter Workflow Manager, select Connections, then Loader to display the Loader Connection Browser.
2. In the Objects list, select Teradata\_Tpump\_Upsert, and then click Edit to display the Connection Object Definition dialog box.
3. Edit the User Name, Password, TDPID, Database Name, Error Database, and Log Table Database, and other attributes as necessary.
4. Repeat these steps for the following objects:
  - Teradata\_Tpump\_Update
  - Teradata\_Tpump\_Insert
  - Teradata\_Tpump\_Delete
  - Teradata\_FastLoad
5. For Unicode environments, append -c UTF8 to the value for the External Loader Executable attribute for each external loader.

For example:

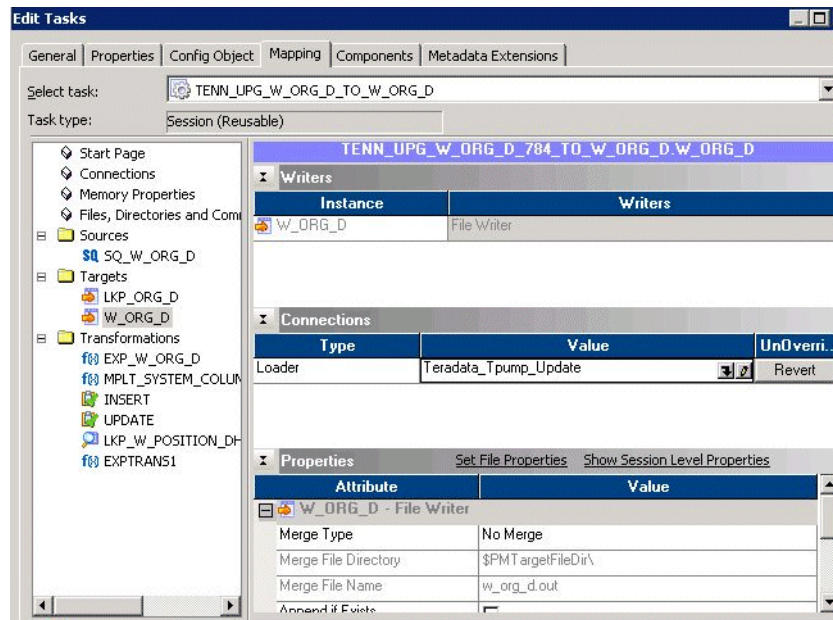
- fastload -c UTF8
- tpump -c UTF8

6. Click Close to close the Loader Connection Browser.

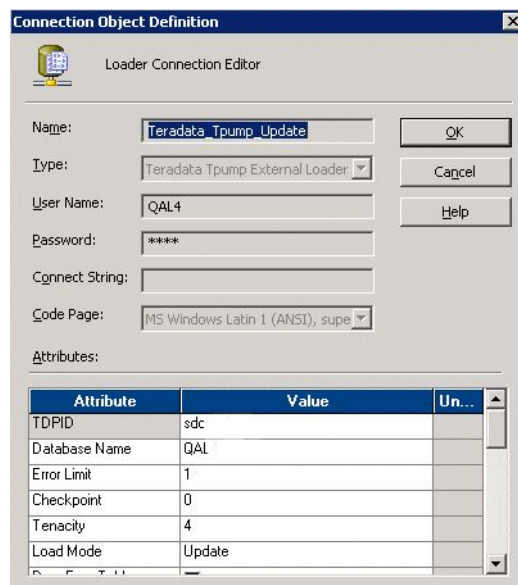
**4.15.2.2.2 How to Specify Teradata Details at the Workflow Level** For each session, you need to specify Teradata details, as follows:



1. In Informatica PowerCenter Workflow Manager, go to the workflow and open the session in the Task Developer pane.
2. Double-click on the session to display the Edit Tasks dialog box.
3. Display the Mapping tab.



4. On the Targets node on the left navigator panel, select the Teradata table name.
5. In the Writers area, select "File Writer" from the Writers drop down list for the target table.
6. In the Connections area, select "Teradata\_Tpump\_Upset" or other "Teradata\_Tpump\_XXX" as the loader.
7. Click on the edit icon (that is, the pencil icon) next to the Connections - Value field to display the Connection Object Definition dialog box.



8. Enter the User Name and Password to log in to the Teradata database.

**Note:** In the Attributes list, make sure that the following attributes are specified:

- TDPID
- Database name
- Error Database
- Log Table Database

If the above attribute values are not inherited from the attributes defined at the connection level, you must specify them.

9. In the Attributes list, specify values for Error Table and Log Table.

## 4.16 Configuring the SiebelUnicodeDB Custom Property

---

**Note:** This procedure is not specific to the Siebel source system. It must be performed for all types of source systems.

---

If your source to target data movement configuration is Unicode to Unicode, you need to create a custom property called SiebelUnicodeDB on Integration Services. If your source to target data movement configuration is either Code Page to Code Page or Code Page to Unicode, you do not need to create this property. For more information on supported source to target configuration modes, and how to determine the source to target configuration for your environment, see [Section 4.3.3, "Code Page and Data Movement Requirements."](#)

### To create and set the SiebelUnicodeDB custom property on Integration Services

1. Log into Informatica PowerCenter Administration Console. For information on how to log into PowerCenter Administration Console, see [Section A.4, "How to Log Into Informatica PowerCenter Administration Console."](#)
2. Select the Integration Service.
3. In the Properties tab, scroll down to the Custom Properties area, and click Edit.
4. In the Name field, enter the following:  
SiebelUnicodeDB
5. In the Value field, enter the following:

[user\_OLTP]@[connectString\_OLTP] [user\_OLAP]@[ConnectString\_OLAP]

Where:

[user\_OLTP] is the database user for the OLTP source database. It must match exactly the value you entered for the User Name field when creating the relational connection for the OLTP in Informatica Workflow Manager.

[ConnectString\_OLTP] is the connect string for the OLTP. It must match exactly the value you entered for the Connect String field when creating the relational connection for the OLTP in Informatica Workflow Manager.

[user\_OLAP] is the database user for the Oracle Business Analytics Warehouse database. It must match exactly the value you entered for the User Name field when creating the relational connection for the data warehouse in Informatica Workflow Manager.

[ConnectionString\_OLAP] is the connect string for the data warehouse. It must match exactly the value you entered for the Connect String field when creating the relational connection for the data warehouse in Informatica Workflow Manager.

For example, `oltp@db204007.host.com olap@db204008.host.com`.

**Note:** Always leave a space between the strings for OLTP and OLAP. You must enter the user names and connection strings in the same case as you used for the relational connections in Informatica Workflow Manager.

For more information about Relational Connections, see [Section 4.15.2, "Configuring Relational Connections in Informatica PowerCenter Workflow Manager"](#).

## 4.17 Setting Up DAC to Receive Email Notification

This section describes how to set up DAC to receive email notification. It includes the following topics:

- [Section 4.17.1, "Configuring Email Recipients in DAC"](#)
- [Section 4.17.2, "Configuring Email in the DAC Server"](#)

### 4.17.1 Configuring Email Recipients in DAC

Follow this procedure to configure email recipients.

#### To configure email recipients in DAC

1. In DAC, navigate to the Setup view.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

2. Click the Email Recipients tab.
3. Click New.
4. In the Edit tab below, enter the following information:

Field	Description
Name	Logical name of the user to be notified.
Email Address	Email address where the notification is sent.
Notification Level	The notification levels are as follows: <ul style="list-style-type: none"> <li>■ 10 -- Notifies recipient of success or failure of each task.</li> <li>■ 5 -- Notifies recipient of success or failure of the entire ETL process.</li> <li>■ 1 -- Notifies recipient that ETL completed successfully.</li> </ul>
Inactive	Indicates whether the selected email activation is active or inactive.
Needs Authentication	Read only value that specifies whether the corporate email server requires authentication (this value is set by choosing Tools, then DAC Server Setup).

## 4.17.2 Configuring Email in the DAC Server

Follow this procedure to configure the email administrator account in DAC, which enables the recipient to receive ETL status updates automatically. For example, the data warehousing administrator might want to be informed when an ETL routine has completed.

The DAC Server has a built-in login-authentication based email (SMTP) client, which connects to any SMTP login-authenticating server.

---

---

**Note:** For the email functionality to work, you must be using an SMTP server in which the SMTP authentication mode LOGIN is enabled. For example, if you are using Microsoft Exchange Server, you must enable the SMTP module and enable authentication in the Basic Mode. The SMTP server outbound email authentication must be turned on for the DAC Server to be able to send email notifications.

---

---

### To configure the email administrator account in the DAC Server

1. In the DAC menu bar, select Tools, then DAC Server Management, then DAC Server Setup.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

A confirmation dialog box asks you to confirm that you want to configure the DAC Repository connection.

2. Click Yes.  
The Server Configuration dialog box appears.
3. Display the Email Configuration tab, and enter the email details for the email address to which you want to send DAC Server information emails.
4. Click Save.

Once the configuration has been completed, you can start the DAC Server.

## 4.18 Additional Configuration Tasks

This section provides configuration steps that may be required depending on your specific environment. All the sections may not be applicable for your environment.

For mandatory, Siebel-specific configuration tasks, see [Section 4.18.3, "Siebel-Specific Configuration Tasks."](#)

For mandatory, Teradata-specific configurations tasks, see [Section 4.18.4, "Teradata-Specific Configuration Tasks."](#)

This section contains the following topics:

- [Section 4.18.1, "Setting DAC Source System Parameters"](#)
- [Section 4.18.2, "Creating Stored Procedures for DB2-UDB"](#)
- [Section 4.18.3, "Siebel-Specific Configuration Tasks"](#)
  - [Section 4.18.3.1, "Migrating Seed Data from a Siebel Source System to the Oracle BI Repository"](#)
  - [Section 4.18.3.2, "Updating Siebel Transactional Database Schema Definitions"](#)

- Section 4.18.4, "Teradata-Specific Configuration Tasks"
  - Section 4.18.4.1, "Setting Up the Code Page File Property for Unicode Environments on Teradata Databases"
  - Section 4.18.4.2, "Setting Up the HOSTS File for Teradata Installations on Windows"
  - Section 4.18.4.3, "Setting Up the HOSTS File for Teradata Installations on UNIX"
  - Section 4.18.4.4, "Teradata-Specific Installation Checklist"

## 4.18.1 Setting DAC Source System Parameters

You set source system parameters (also known as DAC ETL Preferences) in DAC to specify how the ETL routines process data for a container. For example, if operating in the United States, you might set the `$$DLFT_COUNTRY` to 'USA' to identify your data.

---

**Note:** The following preferences are applied to all tasks within a container. If extracting data from more than one source, these preferences will need to be re-applied to each associated container. They can be overridden at the task level by adding the parameter to the specific task and assigning a value there.

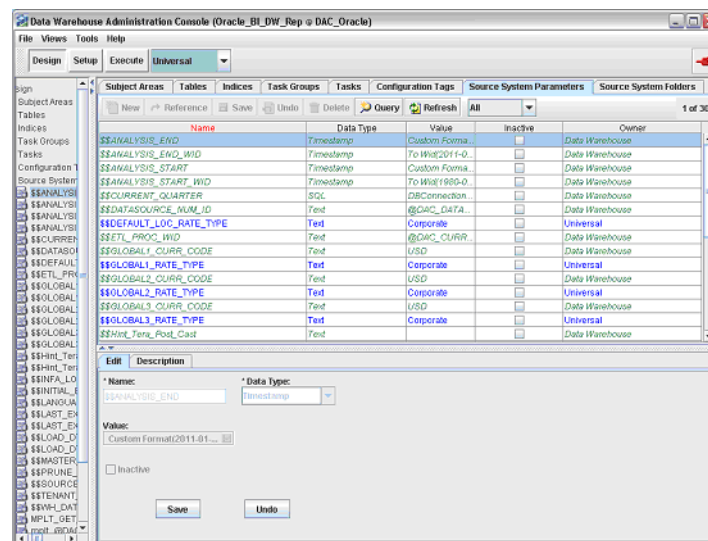
---

### To set DAC source system parameters

1. In DAC, go to the Design view.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)

2. Make sure that you have selected the correct container from the containers drop down list. You cannot edit out-of-the box containers. Make a copy of an existing container in order to make edits.
3. Display the Source System Parameters tab.



---

---

**Note:** For information about recommended settings for specific databases, see [Chapter 3, "Preinstallation and Predeployment Requirements for Oracle BI Applications"](#).

---

---

4. Use the Edit tab below the list of Source System Parameters to change the value of parameters in the list.
5. Click Save.

## 4.18.2 Creating Stored Procedures for DB2-UDB

DAC uses siebstat and siebtrun stored procedures when running ETL processes. Typically, these stored procedures are available in your transactional database; they might not be available on the data warehouse database.

If you need to install the DB2 stored procedures manually, they are located in the installation directory `\OracleBI\dwrep\siebproc\db2udb\`. There is a sub-directory for each platform. For example, stored procedures for the Windows platform are stored in the sub-directory `\OracleBI\dwrep\siebproc\db2udb\win32\`.

Each platform-specific directory contains the following sub-directories:

- `\siebproc\` (containing stored procedures for a 32-bit DB2 environment)
- `\siebproc64\` (containing stored procedures for a 64-bit DB2 environment)

These directories also contain the files `siebproc.sql` and `sqlproc.ksh`, which are used to create the function. For more information, see [Section 4.18.2.2, "How to Create DB2 Stored Procedures"](#).

### 4.18.2.1 How to Verify the Existence of Stored Procedures

This section describes how to verify the existence of stored procedures.

#### To verify the existence of the stored procedures

- From a DB2 command prompt or control center, issue the following SQL scripts:

```
db2 => create table test_siebproc(id int);
DB20000I The SQL command completed successfully.

db2 => insert into test_siebproc(id) values (1);
DB20000I The SQL command completed successfully.

db2 => call siebtrun('TEST_SIEBPROC');
SQL0805N Package "NULLID.SYSSH200" was not found.  SQLSTATE=51002
```

---

---

**Note:** If you get an error message, you do not have the required stored procedures.

---

---

Creating stored procedures must be performed by the database administrator (for more information, see [Section 4.18.2.2, "How to Create DB2 Stored Procedures"](#)).

### 4.18.2.2 How to Create DB2 Stored Procedures

This section describes how to create DB2 stored procedures.

#### To create DB2 stored procedures

1. Copy the DB2 stored procedure directory (i.e. \siebproc\ or \siebproc64\) from the appropriate platform-specific directory to a directory on the DB2 server side.

For example, for a 32-bit DB2 environment on a Windows platform, you might copy the directory \OracleBI\dwrep\siebproc\db2udb\win32\siebproc\ to the directory d:\Program Files\SQLLIB\function\ on the DB2 server side.

---

**Note:** For more information about the location of DB2 stored procedures, see [Section 4.18.2, "Creating Stored Procedures for DB2-UDB"](#).

---

2. If you copied stored procedures for a 64-bit DB2 environment, on the DB2 server side, rename the \siebproc64\ directory to \siebproc\.

For example, if you copied stored procedures to d:\Program Files\SQLLIB\function\siebproc64\, rename this directory to d:\Program Files\SQLLIB\function\siebproc\.

Once these procedures are created, you can verify that they exist. After the test is complete, you can drop the table TEST\_SIEBPROC.

### 4.18.3 Siebel-Specific Configuration Tasks

This section contains tasks you are required to complete if your source system is Siebel. This section contains the following topics:

- [Section 4.18.3.1, "Migrating Seed Data from a Siebel Source System to the Oracle BI Repository"](#)
- [Section 4.18.3.2, "Updating Siebel Transactional Database Schema Definitions"](#)

#### 4.18.3.1 Migrating Seed Data from a Siebel Source System to the Oracle BI Repository

This section describes how to move localized string content from the S\_MSG table in a Siebel OLTP database to the W\_LOCALIZED\_STRING\_G table in the Business Analytics Warehouse database. Use the following steps:

1. Export the customized localization data in the S\_MSG table using the DATAEXP command from a command prompt, as shown in the following sample text:
 

```
DATAEXP /U <Username for OLTP read access> /P <Password> /C
<Siebel_OLTP odbc> /D <Table Owner> /F CUSTSTRINGS.DATA /W Y
/L <EXPCUST.LOG>
```
2. Respond to the prompts as follows:
  - When you are prompted for the TABLE\_NAME to export, enter S\_MSG and press ENTER.
  - When you are prompted for the WHERE CLAUSE for filtering records from the table, enter WHERE LAST\_UPD > '1/1/1980' and press ENTER twice.
3. A message is displayed that indicates the number of rows that were exported. Press ENTER to enter the Database Table Name and to end the process.
4. Import the data into the W\_LOCALIZED\_STRING\_G table using the data file that you generated in the previous step.

Use the /R 1000000 argument in the DATAIMP command to insert the customized strings beyond Row Id 1000000.

For information about importing localization data into the W\_LOCALIZED\_STRING\_G table, see [Section C.1.1, "Upgrading Oracle Business Intelligence Seed Data for Non-English Locales."](#)

5. Import the localization data into the W\_LOCALIZED\_STRING\_G table.

#### 4.18.3.2 Updating Siebel Transactional Database Schema Definitions

---

---

**Note:** The steps in this section are required if are using a Siebel source system.

---

---

This section explains how to update Siebel transactional database schema definitions. It contains the following sections:

- [Section 4.18.3.2.1, "How to Run the DDLIMP Tool From The Command Line"](#)
- [Section 4.18.3.2.2, "How to Apply Siebel CRM Schema Changes For Oracle, DB2/UDB, MSSQL"](#)
- [Section 4.18.3.2.3, "How to Apply CRM Schema Changes for DB2 on OS/390 and z/OS"](#)
- [Section 4.18.3.2.4, "How to Verify the Siebel CRM Schema Changes"](#)
- [Section 4.18.3.2.5, "About Delete Triggers"](#)

**4.18.3.2.1 How to Run the DDLIMP Tool From The Command Line** When you use the DDLIMP utility from a command line to update schema definitions, refer to the following notes:

To run DDLIMP from command line, run the following command:

```
\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC_DSN>
/GSSE_ROLE /F <..\OracleBI\dwrep\DDL_OLTP.CTL> /L
<..\oracleBI\dwrep\DDL_OLTP.log>
```

NOTE: Where <ODBC\_DSN> is the ODBC connection created as described in section [Section 4.3.4.3, "Create ODBC Database Connections"](#).

```
\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC_DSN>
/GSSE_ROLE /W Y /Z Y /F <..\OracleBI\dwrep\DDL_OLTP.CTL>
/L<..\oracleBI\dwrep\DDL_OLTP.log>
```

Additionally you can use the following commands:

/W Y (if the OLTP database is Unicode).

/Z Y (if the OLTP database is DB2 and Unicode).

/B <TABLE\_SPACE\_NAME> if you want to create these table in a separate table space.

/X <INDEX\_TABLE\_SPACE\_NAME> if you want to create the indexes in a separate table space.

/Y Storage File for DB2/390.

**4.18.3.2.2 How to Apply Siebel CRM Schema Changes For Oracle, DB2/UDB, MSSQL** To enable change capture for Oracle's Siebel adapters, you use the ddlimp control file to apply Siebel CRM schema changes, which updates the required image tables in the OLTP.



---



---

**Note:** Using the ddlimp control file to apply schema changes replaces the use of SIF files in Siebel Tools in previous product releases.

---



---

### To apply Siebel CRM schema changes for Oracle, DB2/UDB and MSSQL databases

1. On the machine where Oracle BI Applications is installed, run the following command:

```
..\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC connect string>
/G SSE_ROLE /F <..\OracleBI\dwrep\DDL_OLTP.CTL> /L <..\OracleBI\dwrep\DDL_
OLTP.log>
```

---



---

**Note:** If you are applying schema changes to a Siebel CRM release 6.3, specify DDL\_OLTP\_63.CTL as the /F parameter instead of DDL\_OLTP.CTL.

---



---

For example:

```
DDLIMP /U SADMIN /P SADMIN /C SIEBEL_OLTP /G SSE_ROLE /F
<DRIVE>:\OracleBI\dwrep\DDL_OLTP.CTL /L <DRIVE>:\OracleBI\dwrep\DDL_OLTP.log
```

Notes:

- /P <PASSWORD> - The password for Oracle's CRM OLTP.
- /C <ODBC connect string> - The name of the ODBC connect string.
- For Oracle databases, use the Oracle Merant ODBC Drivers (installed with Oracle BI Applications).
- In addition, you can use the following commands:
  - /W Y - (if the OLTP database is Unicode).
  - /Z Y - (if the OLTP database is DB2 and Unicode or the OLTP database is MS SQL Server and Unicode).
  - /B <TABLE\_SPACE\_NAME> - If you want to create these table in a separate table space.
  - /X <INDEX\_TABLE\_SPACE\_NAME> - If you want to create the indexes in a separate table. space.
  - /Y - Storage File for DB2/390.

2. Restart all servers.

#### 4.18.3.2.3 How to Apply CRM Schema Changes for DB2 on OS/390 and z/OS

1. Edit the following parameters in the Storage control files located in \OracleBI\dwrep\Storeage\_DDL\_OLTP.CTL:

- %1 - Replace with a 2 character database name.
- %indBufPool - Replace it with a index buffer pool name.
- %4kBulPool - Replace it with a 4k TBS buffer pool name.
- %32kBufPool - Replace it with a 32K TBS Buffer Pool name.

2. On the machine where Oracle BI Applications is installed, run the following command:

```

.. \OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC_CSN> >>> /G SSE_
ROLE /F \OracleBI\dwrep\DDL_OLTP_DB2390.CTL> /L <..\oracleBI\dwrep\DDL_
OLTP.log> /5 Y /A <SCHEMA_OWNER> /Y .. \OracleBI\dwrep\Storage_DDL_OLTP.CTL

```

**Notes:**

- /P <PASSWORD> - The password for Oracle's CRM OLTP.
- /C <ODBC connect string> - The name of the ODBC connect string.
- For Oracle databases, use the Oracle Merant ODBC Drivers.
- In addition, you can use the following commands:
  - /W Y - (if the OLTP database is Unicode).
  - /Z Y - (if the OLTP database is DB2 and Unicode or the OLTP database is MS SQL Server and Unicode).
  - /B <TABLE\_SPACE\_NAME> - If you want to create these table in a separate table space.
  - /X <INDEX\_TABLE\_SPACE\_NAME> - If you want to create the indexes in a separate table. space.
  - /Y - Storage File for DB2/390.

3. To create indexes for the tables created, run the following SQL Script from the DB2 command line utility connected to your OLTP Schema:

```
\OracleBI\Dwrep\Create_OLTP_Db2390_index.sql
```

**4.18.3.2.4 How to Verify the Siebel CRM Schema Changes** After applying Siebel CRM schema changes, you need to verify that appropriate tables were created in the transactional database.

**To verify Siebel CRM schema changes**

1. For Siebel Business Applications, make sure that the following tables were created in the transactional database:
  - S\_ETL\_R\_IMG\_1 through S\_ETL\_R\_IMG\_166
  - S\_ETL\_I\_IMG\_1 through S\_ETL\_D\_IMG\_166
  - S\_ETL\_D\_IMG\_1 through S\_ETL\_D\_IMG\_166
  - S\_ETL\_PARAM
  - S\_ETL\_PRD\_ATTR
  - S\_ETL\_PRD\_REL

**4.18.3.2.5 About Delete Triggers**


---

**Note:** Delete triggers are only used with Siebel CRM databases.

---

Delete records in Siebel CRM sources are not propagated to the data warehouse tables. However the mechanism to identify the delete records is provided, as follows:

DAC can create delete triggers on source tables (refer to *Oracle Business Intelligence Data Warehouse Administration Console User Guide*). These triggers write the primary keys of deleted records with the flag D in the corresponding S\_ETL\_I\_IMG table. You need to write a custom SDE extract to pull these deleted primary keys from the image table and take corresponding action on the data warehouse table.

## 4.18.4 Teradata-Specific Configuration Tasks

This section contains tasks you are required to complete if you are using a Teradata database in your Oracle BI Applications environment.

This section contains the following topics:

- [Section 4.18.4.1, "Setting Up the Code Page File Property for Unicode Environments on Teradata Databases"](#)
- [Section 4.18.4.2, "Setting Up the HOSTS File for Teradata Installations on Windows"](#)
- [Section 4.18.4.3, "Setting Up the HOSTS File for Teradata Installations on UNIX"](#)
- [Section 4.18.4.4, "Teradata-Specific Installation Checklist"](#)

### 4.18.4.1 Setting Up the Code Page File Property for Unicode Environments on Teradata Databases

If you have a Unicode environment on a Teradata database, you need to set the code page file property of sessions that use the Teradata external loaders to use the code page UTF8. You need to follow this procedure to set the code page file property for each session that uses a Teradata external loader.

**To set the code page file property for a Unicode environment on a Teradata database**

1. In Informatica PowerCenter Workflow Manager, drag a Workflow into the Workflow Designer pane.
2. In the Workflow Designer pane, double click the Task (for example, SDE\_PSFT\_APTermsDimension) to display the Edit Tasks dialog box.
3. Display the Mappings tab.
4. Select Target in the left pane. In the Properties section, click on the **Set File** link.
5. In the Flat Files - Targets dialog box, in the File Properties area, click Advanced.
6. In the Fixed Width Properties - Targets dialog box, in the Code Page area, select UTF-8 encoding of Unicode.

### 4.18.4.2 Setting Up the HOSTS File for Teradata Installations on Windows

If you are using a Teradata database, you need to set the TDPID parameter on the machine where the Informatica PowerCenter Integration Services service is installed by adding an entry in the HOSTS file.

**To set up the HOSTS file for Teradata installations on Windows**

1. On the machine where the Informatica PowerCenter Integration Services service is installed, go to the `\<SystemRoot>\system32\drivers\etc` directory and open the HOSTS file.
2. In the HOSTS file, enter a line in the following format:

```
<IP address of remote server> <remote server> <remote server>COP<n>
```

For example:

```
172.20.176.208 tdatsvr tdatsvrCOP1
```

where `tdatsvrCOP1` is the alias for the remote server. The alias must begin with an alphabetic string and end with the `COP n` suffix, where `n` is a number between

1 and the total number of applications processors that are associated with the Teradata communications processor.

3. Save the HOSTS file.

For more information about setting the TDPID parameter, see the Teradata documentation.

#### 4.18.4.3 Setting Up the HOSTS File for Teradata Installations on UNIX

If you are using a Teradata database, you need to set the TDPID parameter on the machine where the Informatica Server is installed by adding an entry in the HOSTS file.

##### To set up the HOSTS file for Teradata installations on UNIX

1. How to Set Up the HOSTS File for Teradata Installations on UNIX.
2. In the HOSTS file, enter a line in the following format:

<IP address of remote server> <remote server> <remote server>COP<n>

For example:

```
172.20.176.208 tdatsvr tdatsvrCOP1
```

Where tdatsvrCOP1 is the alias for the remote server. The alias must begin with an alphabetic string and end with the COP<n> suffix, where <n> is a number between 1 and the total number of applications processors that are associated with the Teradata communications processor.

3. Save the HOSTS file.

For more information about setting the TDPID parameter, refer to the Teradata documentation.

#### 4.18.4.4 Teradata-Specific Installation Checklist

Table 4–4 provides a list of Teradata-specific installation and configuration steps that are performed during the Oracle BI Applications installation and configuration process. You should review this list to make sure that you have performed all of the required Teradata-specific steps.

**Table 4–4 Teradata-Specific Installation Checklist for Windows**

Action	Link to Topic
Review Teradata-specific database guidelines.	<a href="#">Section 3.6, "Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse".</a>
Create or drop data warehouse tables.	<a href="#">Section 4.11.2, "Creating Data Warehouse Tables on a Teradata Database".</a>
Create or restore the Informatica Repository using the Informatica Repository Manager. You cannot create or restore the Informatica Repository using DAC.	<a href="#">Section 4.6, "Restoring the Prebuilt Informatica Repository".</a>

**Table 4–4 (Cont.) Teradata-Specific Installation Checklist for Windows**

Action	Link to Topic
Configure the source and data warehouse database relational connections in Informatica PowerCenter Workflow Manager. You cannot use the DAC's Data Warehouse Configurator to configure relational connections.	<a href="#">Section 3.6, "Teradata-Specific Database Guidelines for Oracle Business Analytics Warehouse"</a> .
Set the Code Page File Property for Unicode Environments.	<a href="#">Section 4.18.4.1, "Setting Up the Code Page File Property for Unicode Environments on Teradata Databases"</a>
Set up the HOSTS file.	<a href="#">Section 4.18.4.2, "Setting Up the HOSTS File for Teradata Installations on Windows"</a>

## 4.19 About Running A Full Load ETL

After you have installed and configured Oracle BI Applications, your Oracle Business Analytics Warehouse (OLAP) database is empty. You need to perform a full load ETL to populate your Oracle Business Analytics Warehouse.

The ETL processes for Oracle BI Applications are created and managed in DAC. For detailed information about running ETLs in DAC, see *Oracle Business Intelligence Data Warehouse Administration Console Guide*.

For an example of using DAC to run a full load ETL, see [Section 4.19.1, "An Example of Running a Full Load ETL."](#)

### 4.19.1 An Example of Running a Full Load ETL

This section uses an example to show you how to get started quickly with running a full load ETL. In this example, you have installed Oracle Financial Analytics with an Oracle EBS OLTP data source, and you want to load OLTP data for the subject area Receivables, as follows:

1. In the DAC menu bar, select File, then New Source System Container, to display the New Source System Container dialog box, and specify details of the new container.

**Note:** You cannot make any changes to the preconfigured containers. You must make a copy of a container before you can make any changes to it.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#) For more information about the source system container functionality in DAC, see *Oracle Data Warehouse Administration Console Guide*.

2. Select the Create as a Copy of Existing Container radio button, and select the appropriate container (for example, Oracle 11.5.10) from the Existing Containers drop-down list, then click OK.
3. In the Design view, display the Source System Parameters tab and set the parameters appropriately.
4. In the Setup view, display the DAC System Properties tab and set the properties appropriately.
5. Display the Execute view, and display the Execution Plan tab.

6. Click New, display the Edit tab, and use the Name field to specify a name for the ETL process.
7. Display the Subject Areas tab and click Add/Remove to display the Choose Subject Areas dialog box.
8. Select the new container that you created in step 4 from the container drop down list at the top of the Choose Subject Areas dialog box.
9. Select Financials - Receivables, click Add, then click OK.
10. Display the Parameters tab, and click Generate.
11. On the Parameters tab, edit the parameters as follows:
  - Edit the value of DBConnection\_OLAP and set it to the same value as the name of the OLAP database that you specified in the Physical Data Source dialog box (for example, DataWarehouse).
  - Edit the value of DBConnection\_OLTP and set it to the same value as the name of the OLTP database that you specified in the Physical Data Source dialog box (for example, ORA\_11\_5\_8).
  - If there is a FlatFileConnection parameter, edit the value of FlatFileConnection and set it to the same value as the name of the flat file data source that is specified in the Physical Data Source dialog box (for example, ORA\_11\_5\_8\_Flatfile).
12. On the Execution Plans tab, click Build.
13. On the Execution Plans tab, click Run Now.

DAC will perform a full load for Financials - Receivables.
14. Use the Current Run tab to check the status of the ETL run.

If the ETL run was successful, you will see 'Success' in the Run Status field, and the End Timestamp value will be set to the time and date when the ETL was completed.

If the ETL run fails, use the information in the Description tab and the Audit Trail tab to diagnose the error that caused the failure.

**Tip:** If an ETL run fails, you cannot re-run the ETL until the failed ETL has been cleared from the Current Run tab. To clear an ETL from the Current Run tab, right click on the ETL and select Mark As Completed.

# Part III

---

## Appendices

Part III contains the following sections:

- [Appendix A, "Supporting Tasks for DAC and Informatica PowerCenter"](#)
- [Appendix B, "Configuring Universal Adapter for CRM Analytics"](#)
- [Appendix C, "Localizing Oracle Business Intelligence Deployments"](#)
- [Appendix D, "Integrating Interactive Dashboards and Operational Applications Data"](#)
- [Appendix E, "Configuring Metadata for Analytics Applications"](#)
- [Appendix F, "Oracle Business Analytics Warehouse for Life Sciences Data Considerations"](#)
- [Appendix G, "Pharma Disconnected Analytics Administration - Supplemental Information"](#)
- [Appendix H, "Using Oracle Business Analytics Warehouse Exception Reports"](#)
- [Appendix I, "Troubleshooting Oracle BI Applications"](#)
- [Appendix J, "About the Versioned Informatica Repository"](#)

---

**Note:** For a high-level road map for installation, configuration, and customization steps for Oracle BI Applications, see [Section 2.4, "Roadmap to Installing and Setting Up Oracle BI Applications."](#)

---





---

---

## Supporting Tasks for DAC and Informatica PowerCenter

This section contains additional tasks relating to the DAC Client and DAC Server, and Informatica PowerCenter Client Tools, and contains the following topics:

- [Section A.1, "How to Log into DAC"](#)
- [Section A.2, "How to Set DAC System Properties"](#)
- [Section A.3, "How to Start and Stop the DAC Server"](#)
- [Section A.4, "How to Log Into Informatica PowerCenter Administration Console"](#)
- [Section A.5, "How to Start and Stop Informatica Services"](#)
- [Section A.6, "How to Stop and Start the Informatica Repository Service \(Optional\)"](#)
- [Section A.7, "About the DAC Server Scripts for Bash and C Shells"](#)

### A.1 How to Log into DAC

Before you can log into the DAC, you need to have created a DAC connection, which is a set of stored login details. For information about creating a DAC connection, see [Section 4.10.1, "Logging into DAC."](#)

#### To log into DAC

1. Launch the DAC Client by double-clicking the DAC Client icon on your desktop or navigating to the OracleBI\DAC directory and double-clicking the startclient.bat file.

The Login... dialog is displayed.



2. In the Login... dialog, select a connection for the required DAC Repository from the Connection drop down list.
3. In the Table owner name field, enter the database user name for the DAC Repository database.
4. In the Password field, enter the database password for the DAC Repository database.
5. Click Login to launch the DAC.

The DAC Client launches and connects to the DAC Repository.

For more information about using the DAC Client, see *Oracle Business Intelligence Data Warehouse Administration Console Guide*.

## A.2 How to Set DAC System Properties

This section provides instructions for setting the DAC System Properties.

### To set DAC System Properties

1. Log in to DAC.  
For more information about logging into DAC, see [Section A.1, "How to Log into DAC."](#)
2. From the Views menu, select Setup, then DAC System Properties tab.
3. Set the appropriate values for the following system properties:

Property	Value Required
Analyze Frequency (in days)	For DAC metadata tables, the frequency (in days) the DAC client automatically updates the table and index statistics for the DAC repository. The value must be numerical.
Auto Restart ETL	<p>Possible values are True and False.</p> <p>When set to True: An ETL that is running when the DAC server abnormally terminates will continue running when the DAC server is restarted.</p> <p>When set to False: An ETL that is running when the DAC server abnormally terminates will not automatically restart when the DAC server restarts. The ETL status will be updated to Failed. An administrator will have to manually restart the ETL.</p>

Property	Value Required
DAC Alternate Server Hosts	Host name of the machine where the alternate DAC server resides. The alternate DAC server is used for failover purposes. The DAC client cannot talk to the alternate server unless the main DAC server is not running.
DAC Server Hosts	<p>Host name of the machine where the DAC server resides. You cannot use an IP address for this property.</p> <p>The DAC server and a given DAC repository have a one-to-one mapping. That is, you can only run one DAC server against any given DAC repository. Thus, in the repository you must specify the network host name of the machine where the DAC sever is to be run.</p> <p>This property also takes the value localhost. However, this value is provided for development and testing purposes and should not be used in a production environment.</p>
DAC Server OS	<p>Operating system of the machine where the DAC server resides. Possible values are Windows, Solaris, HP, or AIX.</p> <p>If you move the DAC server from another operating system to AIX, you need to do the following: change the DAC server host to the appropriate value; restart the DAC client; reenter all the password fields for the Informatica servers and database connections; and reconfigure the DAC server on the AIX machine by running <code>serverSetupPrompt.sh</code>.</p>
DAC Server Port	Network port to which the DAC server binds in order to listen to client requests. The default value is 3141. If this port has been assigned to another process, you can enter any numerical port value greater than 1024.
Drop and Create Change Capture Views	<p>Possible values are True and False.</p> <p>When set to True (the default value), the DAC server drops and creates change capture views every time it performs a change capture process, including for both full and incremental loads.</p> <p>Setting this property to True can create system catalog lock up for DB2-UDB and DB2-390 databases. Therefore, by setting the property to False, the DAC server will drop and create views selectively, using the following rules:</p> <ul style="list-style-type: none"> <li>■ In full mode: <ul style="list-style-type: none"> <li>During the change capture phase, views will be dropped and created as full views.</li> <li>During the change capture sync process, incremental views will be generated.</li> </ul> </li> <li>■ In incremental mode: <ul style="list-style-type: none"> <li>If the view exists, it will not be dropped and created.</li> <li>If the view does not exist, the incremental view will be created.</li> </ul> </li> </ul>
Dryrun	<p>Possible values are True and False.</p> <p>Indicates whether tasks are executed without invoking Informatica workflows. The following processes are executed: change capture, truncation of tables, drop and creation of indexes, and analyze statements.</p> <p>This option should be used for debugging purposes only and not used in a production environment.</p>
Generic Task Concurrency LLimit	<p>Determines how many tasks with execution types other than Informatica can be run concurrently. The value must be numerical.</p> <p>To set this value, you should consider what the external tasks do. For example, if the tasks open connections to a database, you should consider how this would affect the preconfigured tasks.</p>

Property	Value Required
HeartBeatInterval	Frequency (in seconds) the DAC server checks on the health of the database connections. The value must be numerical. For example, a value of 300 (the default value) indicates the system will perform subsystem diagnostics and recovery procedures every 300 seconds.
InformaticaFileParameterLocation	Directory where the Informatica parameter file is stored.
Output Redirect	<p>Indicates whether logging information and standard output and errors are redirected to files in the log directory (when property is set to True). The file containing standard output starts with out_ and ends with the .log extension. The standard error messages are in the file starting with err_ and ending with the .log extension.</p> <p>If this property is set to False, the logging information is directed to the machine's standard output and error files, which typically defaults to the console from which the DAC server was launched if the server was launched in a visible console mode. If the server is launched as a Windows service, the logging information is directed to the service log. If the server is launched with the command shell not visible, all logging information is deleted</p>
Repository DB Pool Size	Indicates the maximum number of connections to the DAC repository that the server will maintain.
Scheduler.Poll.Interval	Frequency (in seconds) the DAC server polls for changes in the schedule configuration.
Script After Every ETL	<p>The name of the script or executable to be run after every execution plan.</p> <p>For more information, see the description of the property Script Before Every ETL.</p>
Script Before Every ETL	<p>The name of the script or executable to be run before every execution plan.</p> <p>For example, before running an execution plan, you might want to run a process or perform certain tasks. These can be contained in a script or executable. This file should be placed in the scripts subdirectory of the DAC server.</p> <p>The execution plan runs only after the external process has finished. Therefore, it is important that the script or executable does not fail.</p>
Server Log Level	Output logging level. Possible values are Finest, Finer, Fine, Config, Info, Warning, and Severe. The Severe value produces minimal log details, and Finest produces the most extensive amount of reporting.
SQL Trace	<p>Possible values are True and False.</p> <p>Indicates whether the SQL statements to the DAC repository and database connections are added to the log file. Possible values are True and False. The True value sends a hint to the database connectivity layer of the DAC server to enable SQL tracing; thus, every SQL statement that is run by the DAC server is spooled to the appropriate output log file.</p>

### A.3 How to Start and Stop the DAC Server

This section explains how to start and stop the DAC Server.

#### To start or stop the DAC Server on Windows

1. If you installed the DAC Server with the Oracle BI Applications installation, choose the Windows Start menu, then Programs, then Oracle Business Intelligence, then Oracle DAC, and then Start DAC Server or Stop DAC Server.

2. If you installed the DAC Server by copying the \DAC\ directory to a machine, run the \DAC\startserver.bat script or \DAC\stopserver.bat script.

#### To start the DAC Server on UNIX and Linux

1. In csh-related shells, issue the following command:

```
./startserver.csh  
On AIX, use startserver_aix.csh.
```

2. In bash-related shells, issue the following command:

```
./startserver.sh  
On AIX, use startserver_aix.sh.
```

#### To run the DAC Server in the background

1. In csh-related shells, issue the following command:

```
nohup startserver.csh 2>&1 &
```

2. In bash-related shells, issue the following command:

```
nohup startserver.sh 2>&1 &
```

The nohup command allows the DAC Server to run in the background on UNIX even if the DAC Client is disconnected. **Note:** To stop DAC Server running as a background process, use stopserver.sh or stopserver.csh.

#### To stop the DAC Server on UNIX and Linux

1. In csh-related shells, issue the following command:

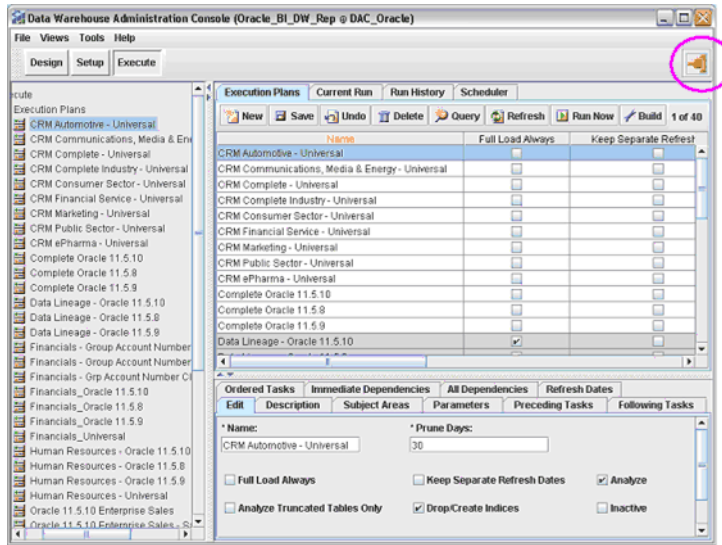
```
./stopserver.csh
```

2. In bash-related shells, issue the following command:

```
./stopserver.sh
```

**Note:** When you execute ./stopserver.csh or ./stopserver.sh, the server will issue a warning about the shutdown request. When the server receives the request, it will shut down even if there is an ETL in progress. The statuses for the ETL run will not be set correctly. The next time the DAC Server starts, it will set the status to Failed for any uncompleted run.

**Tip:** When you start the DAC Server, look at the DAC Server status icon in the DAC console to make sure that the DAC Server has started. The DAC Server status icon should either be orange (idle) or green (active). The screen shot below shows the DAC Server status icon highlighted.



## A.4 How to Log Into Informatica PowerCenter Administration Console

Informatica PowerCenter Administration Console is installed on the machine that hosts the gateway node for the PowerCenter domain. For a single machine install of PowerCenter Services as described in this chapter, PowerCenter Administration Console is installed along with PowerCenter Services.

To log into PowerCenter Administration Console

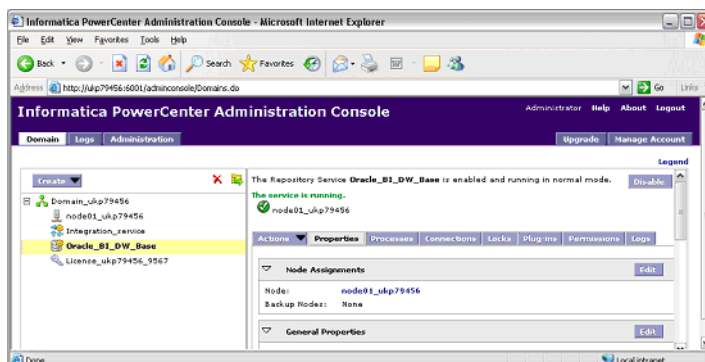
1. In Microsoft Internet Explorer or Mozilla Firefox, access the following URL:

`http://<gateway host>:<domain port>`

Where:

- `<gateway host>` is the name of the machine on which the gateway node has been configured; for a single-machine installation of PowerCenter Services it is the name of the machine on which PowerCenter Services has been installed.
  - `<domain port>` is the port number for the gateway. The default port number is 6001.
2. In the login page, enter the domain username and password that was specified when you created the domain during installation of PowerCenter Services.

For more information about logging into PowerCenter Administration Console, refer to the Appendix in the *Informatica PowerCenter Installation and Configuration Guide*.



## A.5 How to Start and Stop Informatica Services

After you have installed Informatica PowerCenter, you might need to start the Informatica services manually. The Informatica services must be running to enable you to access the Oracle Business Analytics Warehouse using the Informatica tools.

### On Windows

1. Start and stop the Informatica Services from the Start Menu > Programs > Informatica PowerCenter 8.1.1 > Services > Start Informatica Services or Stop Informatica Services.

You can also start and stop Informatica Services from the Windows Services panel.

### On UNIX or Linux

1. Use the `infaservice` command to start or stop Informatica Services, as follows:

```
Infaservice [startup | shutdown]
```

The `infaservice` command is located in  
`$Informatica/PowerCenter8.1.1/server/tomcat/bin.`

The Informatica Services may also be stopped on a node by shutting down the node using PowerCenter Administration Console.

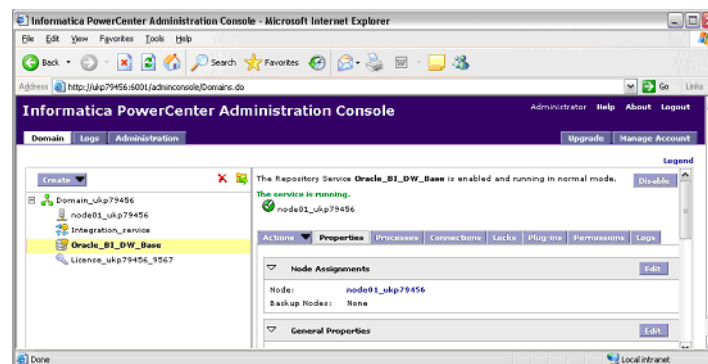
For more information about stopping and starting Informatica PowerCenter Services, refer to the Appendix in the *Informatica PowerCenter Installation and Configuration Guide*.

## A.6 How to Stop and Start the Informatica Repository Service (Optional)

This section explains how to stop and start the Informatica Repository Service. For example, you might need to re-start the service after making changes to the service properties.

### To stop and start the Informatica Repository

1. In Informatica PowerCenter Administration Console, in the Domain tab, select the Repository Service (for example, `Oracle_BI_DW_Base`).



2. Do one of the following:
  - To stop the Informatica Repository Service, click Disable.
  - To start the Informatica Repository Service, click Enable.

## A.7 About the DAC Server Scripts for Bash and C Shells

Shell scripts are provided in the \*.sh and \*.csh formats. You must use the appropriate scripts in the appropriate shells, because their syntaxes are different. [Table A-1](#) lists the available shell scripts and their usage. These files contain comments that provide information about how to configure the scripts.

**Table A-1 Oracle Business Analytics Warehouse Scripts for Bash and C Shells**

Script	Purpose
config.sh config.csh	Configures the environment variables for DAC_HOME and JAVA_HOME.
dacCmdLine.sh dacCmdLine.csh	Invokes DAC commands on the DAC Server. See the file for usage information.
dac_env.sh dac_env.csh	Configures environment variables for the DAC Server.
serverSetupPrompt.sh serverSetupPrompt.csh	Configures DAC metadata repository connection information.
serverinfo.sh serverinfo.csh	Displays the DAC Server version.
startserver.sh startserver.csh	Starts the DAC Server on machines running Solaris or Linux, and HP.
startserver_aix.sh startserver_aix.csh	Starts the DAC Server on AIX machines.
stopserver.sh stopserver.csh	Shuts down the DAC Server.

---

**Note:** The files listed in [Table A-1](#) need to have read, write, and execute permissions. If the files do not have these permissions, modify them using the `chmod` command.

When you copy across these files, use a MS-DOS to UNIX conversion tool, convert the script files to UNIX format (that is, remove the carriage return and line feed characters). There are many MS-DOS to UNIX conversion tools that are freely available for download on the Internet. Alternatively, you can manually remove the carriage return and line feed characters from the script files.

On some shells, `export` commands in `dac_env.sh` and `config.sh` might not work correctly. In this case, try breaking the commands in two. For example, change the command from:

```
export JAVA_HOME=/opt/java1.5
```

To:

```
JAVA_HOME=/opt/java1.5
export JAVA_HOME
```

---

Follow these procedures to start and stop the DAC Server on UNIX. Before you can start the DAC Server, you must have already configured the `config.csh` or `config.sh` and `dac_env.csh` or `dac_env.sh` files.



**To start the DAC Server**

1. In csh-related shells, issue the following command:

```
./startserver.csh
```

2. In bash-related shells, issue the following command:

```
./startserver.sh
```

**To run the DAC Server in the background**

1. In csh-related shells, issue the following command:

```
nohup startserver.csh 2>&1 &
```

2. In bash-related shells, issue the following command:

```
nohup startserver.sh 2>&1 &
```

The nohup command allows the DAC Server to run in the background on UNIX even if the DAC Client is disconnected.

**To stop the DAC Server**

1. In csh related shells, issue the following command:

```
./stopserver.csh
```

2. In bash-related shells, issue the following command:

```
./stopserver.sh
```

---



---

**Note:** When you execute `./stopserver.csh` or `./stopserver.sh`, the server will issue a warning about the shutdown request. When the server receives the request, it will shut down even if there is an ETL in progress. The statuses for the ETL run will not be set correctly. The next time the DAC Server starts, it will set the status to Failed for any uncompleted run.

---



---

## A.8 How to Configure the DAC Metadata Repository

This section explains how to configure the DAC metadata repository, and includes the following topics:

- [Section A.8.1, "How to Set Up DAC System Properties"](#)
- [Section A.8.2, "How to Activate Join Indices for Teradata Databases"](#)

### A.8.1 How to Set Up DAC System Properties

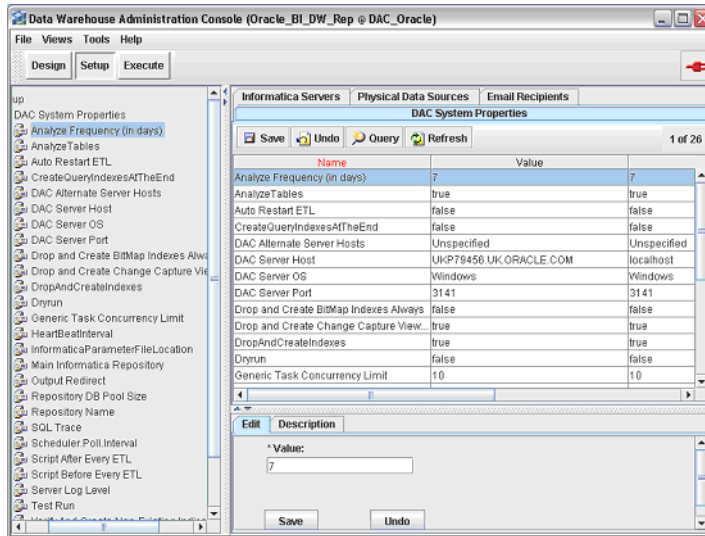
Follow this procedure to set up the DAC system properties, which determine the behavior of the DAC Server. If you do not change a property value, Oracle BI Applications uses the property value that is automatically set during the Oracle BI Applications installation.

**To set up the DAC system properties**

1. In DAC, display the Setup view.

For more information about logging into DAC, see [Section A.1, "How to Log into DAC"](#).

2. Display the DAC System Properties tab.



3. If required, change the property values.

**Note:** Possible values for all properties have to be keyed in exactly as described in the description tab (at the bottom) of each property. For example Auto Restart has possible values of false and true (case sensitive), and Server Log Level has the following possible values SEVERE, FINEST, FINER, FINE, INFO, WARNING (case sensitive).

Property	Description
Analyze Frequency (in days)	For DAC metadata tables, the frequency (in days) the DAC Client automatically updates the table and index statistics for the DAC repository. The value must be numerical.
Auto Restart ETL	<p>Possible values are TRUE and FALSE.</p> <p>When set to TRUE: An ETL that is running when the DAC Server abnormally terminates will continue running when the DAC Server is restarted.</p> <p>When set to FALSE: An ETL that is running when the DAC Server abnormally terminates will not automatically restart when the DAC Server restarts. The ETL status will be updated to Failed. An administrator will have to manually restart the ETL.</p>
CreateQueryIndexesAtTheEnd	<p>Possible values are True and False.</p> <p>During the ETL process, the DAC Server automatically drops and creates indices. When set to True, this property groups all indices of the Query type and creates them after the ETL is complete.</p> <p>The DropAndCreateIndexes property takes precedence over this property. Therefore, if the DropAndCreateIndexes property is set to False, you cannot set the property CreateQueryIndexesAtTheEnd to True to have indices of the Query type created at the end of the ETL process. Also, be aware that when this property is set to True, tables will be analyzed twice, (once at the end of the ETL indexes and once at the end of query indexes).</p> <p>If any indices are marked as Query type indices, and are used by ETL processes, it can adversely affect the performance of the ETL process.</p>
DAC Alternate Server Hosts	The failover server for the DAC.

Property	Description
DAC Server Host	<p>Host name of the machine where the DAC Server resides. You cannot use an IP address for this property.</p> <p><b>Note:</b> The DAC Server and a given DAC repository have a one-to-one mapping. That is, you can only run one DAC Server against any given DAC repository. Thus, in the repository you must specify the network host name of the machine where the DAC sever is to be run.</p> <p><b>Note:</b> This property also takes the value <i>localhost</i>. However, this value is provided for development and testing purposes and should not be used in a production environment.</p>
DAC Server OS	<p>Operating system of the machine where the DAC Server resides. Possible values are Windows, Solaris, HP, AIX, Linux.</p> <p><b>Note:</b> If you move the DAC Server from another operating system to AIX, you need to do the following: change the DAC Server host to the appropriate value; restart the DAC Client; re-enter all the password fields for the Informatica PowerCenter Integration Services and database connections; and reconfigure the DAC Server on the AIX machine by running <code>serverSetupPrompt.sh</code>.</p>
DAC Server Port	<p>Network port to which the DAC Server binds in order to listen to client requests. The default value is 3141. If this port has been assigned to another process, you can enter any numerical port value greater than 1024.</p>
Drop and Create Change Capture Views Always	<p>Possible values are True and False.</p> <p>When set to True (the default value), the DAC Server drops and creates change capture views every time it performs a change capture process, including for both full and incremental loads.</p> <p>Setting this property to True can create system catalog lock up for DB2-UDB and DB2-390 databases. Therefore, by setting the property to False, the DAC Server will drop and create views selectively.</p> <p><b>Note:</b> This setting only applies to Siebel Adapters. Change Capture Views are built on the Siebel (CRM) or Oracle E-Business Suite OLTP database.</p>
Dryrun	<p>Possible values are True and False.</p> <p>Indicates whether tasks are executed without invoking Informatica workflows. The following processes are executed: change capture, truncation of tables, drop and creation of indices, and analyze statements. Possible values are True and False.</p> <p><b>Note:</b> This option should be used for debugging purposes only and not used in a production environment.</p>
Generic Task Concurrency Limit	<p>Determines how many tasks with execution types other than Informatica can be run concurrently. The value must be numerical.</p> <p>To set this value, you should consider what the external tasks do. For example, if the tasks open connections to a database, you should consider how this would affect the preconfigured tasks.</p>
HeartBeatInterval	<p>Frequency (in seconds) the DAC Server checks on the health of the database connections. The value must be numerical. For example, a value of 300 (the default value) indicates the system will perform subsystem diagnostics and recovery procedures every 300 seconds.</p>
InformaticaParameter FileLocation	<p>This file location can be any file system location that is accessible to both the Informatica PowerCenter Integration Services service and the DAC Server. It is the same location that is entered as the Source File Directory in Informatica PowerCenter Workflow Manager.</p> <p><b>Note:</b> This file location also should not have spaces.</p>

Property	Description
Output Redirect	<p>Indicates whether logging information and standard output and errors are redirected to files in the log directory (when property is set to True). The file containing standard output starts with out_ and ends with the .log extension. The standard error messages are in the file starting with err_ and ending with the .log extension.</p> <p>If this property is set to False, the logging information is directed to the machine's standard output and error files, which typically defaults to the console from which the DAC Server was launched if the server was launched in a visible console mode.</p>
Repository DB Pool Size	<p>Maximum number of connections to the DAC repository the DAC Server will maintain. The value must be numerical.</p> <p><b>Note:</b> As the DAC Server runs, it needs to perform transactions and queries on the DAC repository. In order for it to do so efficiently, it internally maintains a pool of preexisting database connections. This pool of database connections reduces latency and leads to better use of resources through connection sharing. However, you need to specify the maximum size of the connection pool. This setting should be determined in consultation with the database administrator of the database where the DAC repository resides.</p> <p><b>Note:</b> This property was deprecated as of version 7.7.1.5.</p>
Repository Name	Unique name for the DAC repository (for example DAC).
Scheduler.Poll.Interval	Frequency (in seconds) the DAC Server polls for changes in the schedule configuration.
Script After Every ETL	<p>The name of the script or executable to be run after every execution plan.</p> <p>For more information, see the description of the property Script Before Every ETL.</p>
Script Before Every ETL	<p>The name of the script or executable to be run before every execution plan.</p> <p>For example, before running an execution plan, you might want to run a process or perform certain tasks. These can be contained in a script or executable. This file should be placed in the scripts subdirectory of the DAC Server.</p> <p>The execution plan runs only after the external process has finished. Therefore, it is important that the script or executable does not fail.</p>
Server Log Level	Output logging level. Possible values are Finest, Finer, Fine, Config, Info, Warning, and Severe. The Severe value produces minimal log details, and Finest produces the most extensive amount of reporting.
SQL Trace	<p>Possible values are True and False.</p> <p>Indicates whether the SQL statements to the DAC repository and database connections are added to the log file. Possible values are True and False. The True value sends a hint to the database connectivity layer of the DAC Server to enable SQL tracing; thus, every SQL statement that is run by the DAC Server is spooled to the appropriate output log file.</p> <p>It is recommended that you set this property to False.</p>

Property	Description
Verify and Create Non-Existing Indices	<p>Possible values are True and False.</p> <p>Indicates whether indices defined in the DAC repository will be automatically created in the data warehouse database during an incremental load.</p> <p><b>Note:</b> When this system property is set to True, the DAC Server verifies whether indices defined in the DAC repository are also defined in the data warehouse database. This verification process can delay the execution of an execution plan.</p>

## A.8.2 How to Activate Join Indices for Teradata Databases

For Teradata databases, the preconfigured tasks for creating and dropping join indices are inactive.

### To activate join indices for Teradata databases

1. Create a new execution plan with the list of subject areas that you are interested for the ETL.
2. Query for all tasks whose name starts with 'Teradata Drop' and add them as preceding tasks.
3. Query for all tasks whose name start with 'Teradata Create' and add them as following tasks.
4. Assemble the execution plan parameters in the Parameters tab and configure the parameters.
5. Redesign the execution plans.



---

---

## Configuring Universal Adapter for CRM Analytics

This appendix provides instructions for Configuring Universal Adapter for CRM Analytics, and contains the following topics:

- [Section B.1, "What is Universal Adapter for CRM Analytics?"](#)
- [Section B.2, "Data Format Requirements"](#)
- [Section B.3, "About the Sample Source Files Provided"](#)
- [Section B.4, "How to Create Data Files for Universal Adapter for CRM Analytics"](#)
- [Section B.5, "List of Sample Source Files for Universal Adapter for CRM Analytics"](#)

### B.1 What is Universal Adapter for CRM Analytics?

Universal Adapter for CRM Analytics enables you to extract data from CRM transactional applications for analysis, when pre-packaged business adapters for these sources are not available. Unlike the pre-packaged business adapters that can only be used for particular source systems, Universal Adapter for CRM Analytics can be used for any source systems as long as the source data can be presented in the specified flat file format.

The CRM source data could also reside in custom applications or legacy systems. Universal Adapter for CRM Analytics is shipped as part of any of the Oracle BI Applications CRM Analytics product and is not separately priced.

### B.2 Data Format Requirements

To use Universal Adapter for CRM Analytics, source data should be presented in flat files according to the following specifications:

- Data should be in comma delimited format (\*.csv files).
- All date values should be in the format of YYYYMMDDHH24MISS. For example, 20071231140300 should be used for December 31, 2007, 2:03 pm.
- If a string data value contains one or more commas, then it should be double quoted.
- Data should start from line six of each file. The first five lines of each file will be skipped during ETL process.





- [Section B.5.2, "List of Source Files for Specific Subject Areas"](#)

## B.5.1 List of Source Files for Seed Data and Common Dimensions

This section lists source files for seed data and common dimensions. These source files are mandatory and should be populated when using the CRM Universal Adapter.

**Table B-1 Seed Data and Common Dimension Source Files**

CRM Subject Area	Associated Source Files
Seed Data	<ul style="list-style-type: none"> <li>■ file_codes_common</li> <li>■ file_costlst_g</li> <li>■ file_dual_g</li> <li>■ file_exch_rate</li> <li>■ file_lov_d</li> <li>■ file_lst_of_val_g</li> </ul>
Common Dimension	<ul style="list-style-type: none"> <li>■ file_employee</li> <li>■ file_geo_ds</li> <li>■ file_int_org</li> <li>■ file_int_org_dh</li> <li>■ file_org</li> <li>■ file_person</li> <li>■ file_position_ds</li> <li>■ file_product</li> <li>■ file_user</li> </ul>

## B.5.2 List of Source Files for Specific Subject Areas

This section lists CRM subject areas and their related source files. Populate these source files selectively, depending on your business needs.

**Table B-2 Subject Areas and Associated Source Files**

CRM Subject Area	Associated Source Files
CRM - Forecasting	<ul style="list-style-type: none"> <li>■ file_dtl_fcst_fs</li> <li>■ file_fcst_ser_ds</li> <li>■ file_industry_ds</li> <li>■ file_opty_ds</li> <li>■ file_source</li> <li>■ file_sum_fcst_fs</li> <li>■ file_terr</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

CRM Subject Area	Associated Source Files
CRM - Marketing	<ul style="list-style-type: none"> <li>▪ file_camp_hist_fs</li> <li>▪ file_camp_opty_fs</li> <li>▪ file_geo_region_ds</li> <li>▪ file_household_ds</li> <li>▪ file_household_fs</li> <li>▪ file_industry_ds</li> <li>▪ file_ld_wave_ds</li> <li>▪ file_mktg_cost_fs</li> <li>▪ file_mktg_goal_fs</li> <li>▪ file_offer_ds</li> <li>▪ file_ofr_prod_fs</li> <li>▪ file_opty_ds</li> <li>▪ file_period_ds</li> <li>▪ file_person_fs</li> <li>▪ file_regn_ds</li> <li>▪ file_resp_prod_fs</li> <li>▪ file_response_ds</li> <li>▪ file_response_fs</li> <li>▪ file_segment_ds</li> <li>▪ file_source</li> <li>▪ file_terr</li> <li>▪ file_vendor_ds</li> <li>▪ file_wave_ds</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

CRM Subject Area	Associated Source Files
CRM - Sales	<ul style="list-style-type: none"> <li>▪ file_agree_ds</li> <li>▪ file_entlmnt_ds</li> <li>▪ file_geo_region_ds</li> <li>▪ file_industry_ds</li> <li>▪ file_invoice_fs</li> <li>▪ file_offer_ds</li> <li>▪ file_oiprdatr_ds</li> <li>▪ file_opty_cmpt_fs</li> <li>▪ file_opty_con_fs</li> <li>▪ file_opty_ds</li> <li>▪ file_opty_stg_fs</li> <li>▪ file_order_ds</li> <li>▪ file_orderitem_fs</li> <li>▪ file_person_fs</li> <li>▪ file_prdatrnm_ds</li> <li>▪ file_qiprdatr_ds</li> <li>▪ file_quote_ds</li> <li>▪ file_quoteitem_fs</li> <li>▪ file_regn_ds</li> <li>▪ file_rev_n_fs</li> <li>▪ file_segment_ds</li> <li>▪ file_source</li> <li>▪ file_srvreq_ds</li> <li>▪ file_sstage_ds</li> <li>▪ file_terr</li> </ul>
CRM - Service	<ul style="list-style-type: none"> <li>▪ file_activity_fs</li> <li>▪ file_agree_ds</li> <li>▪ file_agreeitem_fs</li> <li>▪ file_assess_ds</li> <li>▪ file_asset_ds</li> <li>▪ file_asset_fs</li> <li>▪ file_claimmnt_ds</li> <li>▪ file_entlmnt_ds</li> <li>▪ file_opty_ds</li> <li>▪ file_quote_ds</li> <li>▪ file_region_ds</li> <li>▪ file_source</li> <li>▪ file_srvreq_ds</li> <li>▪ file_srvreq_fs</li> <li>▪ file_survey_fs</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

<b>CRM Subject Area</b>	<b>Associated Source Files</b>
CRM - Automotive Vehicle Sales	<ul style="list-style-type: none"> <li>■ file_asset_ds</li> <li>■ file_household_ds</li> <li>■ file_vehicle_fs</li> </ul>
CRM - Comm, Media and Energy - Quotes and Orders	<ul style="list-style-type: none"> <li>■ file_agree_ds</li> <li>■ file_entlmnt_ds</li> <li>■ file_industry_ds</li> <li>■ file_offer_ds</li> <li>■ file_oiprdatr_ds</li> <li>■ file_opty_ds</li> <li>■ file_order_ds</li> <li>■ file_orderitem_fs</li> <li>■ file_prilstitem_ds</li> <li>■ file_qiprdatr_ds</li> <li>■ file_quote_ds</li> <li>■ file_quoteitem_fs</li> <li>■ file_regn_ds</li> <li>■ file_segment_ds</li> <li>■ file_source</li> <li>■ file_srvreq_ds</li> <li>■ file_sstage_ds</li> <li>■ file_terr</li> </ul>
CRM - Consumer Sector Fund	<ul style="list-style-type: none"> <li>■ file_fndhdr_fs</li> <li>■ file_fndtxn_fs</li> <li>■ file_fund_ds</li> <li>■ file_fund_fs</li> <li>■ file_plan_promo_ds</li> <li>■ file_terr</li> </ul>
CRM - Consumer Sector - Performance	<ul style="list-style-type: none"> <li>■ file_performance_fs</li> </ul>
CRM - Consumer Sector - Promotion	<ul style="list-style-type: none"> <li>■ file_plan_promo_ds</li> <li>■ file_promotion_fs</li> </ul>
CRM - Consumer Sector - Retail Audit	<ul style="list-style-type: none"> <li>■ file_invloc_ds</li> <li>■ file_plan_promo_ds</li> <li>■ file_rtlaudit_fs</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

<b>CRM Subject Area</b>	<b>Associated Source Files</b>
CRM - ePharma - Account Call	<ul style="list-style-type: none"> <li>▪ file_accnt_rnk_ds</li> <li>▪ file_affl_aa_hs</li> <li>▪ file_affl_ac_hs</li> <li>▪ file_affl_ca_hs</li> <li>▪ file_affl_cc_hs</li> <li>▪ file_area_ds</li> <li>▪ file_call_accnt_fs</li> <li>▪ file_call_attnd_fs</li> <li>▪ file_con_rnk_ds</li> <li>▪ file_postn_con_ds</li> </ul>
CRM - ePharma - Contact Call	<ul style="list-style-type: none"> <li>▪ file_affl_ac_hs</li> <li>▪ file_affl_cc_hs</li> <li>▪ file_area_ds</li> <li>▪ file_call_con_fs</li> <li>▪ file_con_rnk_ds</li> <li>▪ file_postn_con_ds</li> </ul>
CRM - ePharma - Direct Sales	<ul style="list-style-type: none"> <li>▪ file_accnt_rnk_ds</li> <li>▪ file_affl_aa_hs</li> <li>▪ file_affl_ca_hs</li> <li>▪ file_alignmt_dhs</li> <li>▪ file_area_ds</li> <li>▪ file_ins_plan_ds</li> <li>▪ file_syndd_ds_fs</li> </ul>
CRM - ePharma - Indirect Sales	<ul style="list-style-type: none"> <li>▪ file_accnt_rnk_ds</li> <li>▪ file_affl_aa_hs</li> <li>▪ file_affl_ca_hs</li> <li>▪ file_alignmt_dhs</li> <li>▪ file_area_ds</li> <li>▪ file_ins_plan_ds</li> <li>▪ file_syndd_ids_fs</li> </ul>
CRM - ePharma - Medical Education	<ul style="list-style-type: none"> <li>▪ file_area_ds</li> <li>▪ file_con_rnk_ds</li> <li>▪ file_med_att_tmp</li> <li>▪ file_med_cst_tmp</li> <li>▪ file_med_ed_fs</li> <li>▪ file_med_evt_ds</li> <li>▪ file_med_plan_ds</li> <li>▪ file_postn_con_ds</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

CRM Subject Area	Associated Source Files
CRM - ePharma - Objective	<ul style="list-style-type: none"> <li>■ file_accnt_rnk_ds</li> <li>■ file_affl_aa_hs</li> <li>■ file_affl_ac_hs</li> <li>■ file_affl_ca_hs</li> <li>■ file_affl_cc_hs</li> <li>■ file_alignmt_dhs</li> <li>■ file_area_ds</li> <li>■ file_call_accnt_fs</li> <li>■ file_call_con_fs</li> <li>■ file_con_rnk_ds</li> <li>■ file_ins_plan_ds</li> <li>■ file_objective_fs</li> <li>■ file_plan_promo_ds</li> <li>■ file_postn_con_ds</li> <li>■ file_syndd_ds_fs</li> <li>■ file_syndd_ids_fs</li> <li>■ file_syndd_rx_fs</li> </ul>
CRM - ePharma - Physician Plan Prescription	<ul style="list-style-type: none"> <li>■ file_affl_aa_hs</li> <li>■ file_affl_ac_hs</li> <li>■ file_affl_ca_hs</li> <li>■ file_affl_cc_hs</li> <li>■ file_alignmt_dhs</li> <li>■ file_area_ds</li> <li>■ file_con_rnk_ds</li> <li>■ file_postn_con_ds</li> <li>■ file_syndd_xpt_fs</li> </ul>
CRM - ePharma - Physician Prescription	<ul style="list-style-type: none"> <li>■ file_affl_ac_hs</li> <li>■ file_affl_cc_hs</li> <li>■ file_alignmt_dhs</li> <li>■ file_area_ds</li> <li>■ file_con_rnk_ds</li> <li>■ file_ins_plan_ds</li> <li>■ file_postn_con_ds</li> <li>■ file_syndd_rx_fs</li> </ul>
CRM - ePharma - Plan Prescription	<ul style="list-style-type: none"> <li>■ file_affl_aa_hs</li> <li>■ file_affl_ca_hs</li> <li>■ file_alignmt_dhs</li> <li>■ file_area_ds</li> <li>■ file_syndd_pt_fs</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

CRM Subject Area	Associated Source Files
CRM - ePharma - Weekly Physician Prescription	<ul style="list-style-type: none"> <li>■ file_affl_ac_hs</li> <li>■ file_affl_cc_hs</li> <li>■ file_alignmt_dhs</li> <li>■ file_area_ds</li> <li>■ file_con_rnk_ds</li> <li>■ file_ins_plan_ds</li> <li>■ file_postn_con_ds</li> <li>■ file_syndd_w_fs</li> </ul>
CRM - ePharma - Profile Rank	<ul style="list-style-type: none"> <li>■ file_accnt_rnk_ds</li> <li>■ file_affl_aa_hs</li> <li>■ file_affl_ac_hs</li> <li>■ file_affl_ca_hs</li> <li>■ file_affl_cc_hs</li> <li>■ file_area_ds</li> <li>■ file_con_rnk_ds</li> <li>■ file_per_rank_fs</li> <li>■ file_postn_con_ds</li> </ul>
CRM - Financial Service - Accounts and Application	<ul style="list-style-type: none"> <li>■ file_agree_ds</li> <li>■ file_asset_ds</li> <li>■ file_asset_fs</li> <li>■ file_balhstry_fs</li> <li>■ file_fn_holding_ds</li> <li>■ file_fn_holding_fs</li> <li>■ file_industry_ds</li> <li>■ file_ins_claim_ds</li> <li>■ file_offer_ds</li> <li>■ file_opty_ds</li> <li>■ file_region_ds</li> <li>■ file_regn_ds</li> <li>■ file_revn_fs</li> <li>■ file_segment_ds</li> <li>■ file_source</li> <li>■ file_src_offr_hs</li> <li>■ file_srvreq_ds</li> <li>■ file_srvreq_fs</li> <li>■ file_sstage_ds</li> <li>■ file_terr</li> </ul>

**Table B-2 (Cont.) Subject Areas and Associated Source Files**

<b>CRM Subject Area</b>	<b>Associated Source Files</b>
CRM - Financial Service - Accounts and Insurances	<ul style="list-style-type: none"> <li>■ file_asset_ds</li> <li>■ file_asset_fs</li> <li>■ file_claimmnt_ds</li> <li>■ file_industry_ds</li> <li>■ file_ins_claim_ds</li> <li>■ file_ins_claim_fs</li> <li>■ file_terr</li> </ul>
CRM - Financial Service - Balance History	<ul style="list-style-type: none"> <li>■ file_asset_ds</li> <li>■ file_balhstry_fs</li> <li>■ file_industry_ds</li> </ul>
CRM - Public Sector - Benefit	<ul style="list-style-type: none"> <li>■ file_case_ds</li> <li>■ file_order_ds</li> <li>■ file_orderitem_fs</li> </ul>
CRM - Public Sector - Case	<ul style="list-style-type: none"> <li>■ file_activity_fs</li> <li>■ file_case_ds</li> <li>■ file_case_fs</li> <li>■ file_incident_ds</li> <li>■ file_lead_ds</li> <li>■ file_period_ds</li> <li>■ file_region_ds</li> <li>■ file_srvreq_ds</li> <li>■ file_srvreq_fs</li> </ul>
CRM - Public Sector - Incident	<ul style="list-style-type: none"> <li>■ file_case_ds</li> <li>■ file_incident_ds</li> <li>■ file_incident_fs</li> <li>■ file_period_ds</li> </ul>
CRM - Public Sector - Lead	<ul style="list-style-type: none"> <li>■ file_case_ds</li> <li>■ file_lead_ds</li> <li>■ file_lead_fs</li> <li>■ file_period_ds</li> </ul>



---

---

# Localizing Oracle Business Intelligence Deployments

Oracle Business Intelligence is designed to allow users to dynamically change their preferred language and locale preferences. This chapter contains the following topics on how to configure Oracle Business Intelligence Applications for deployment in one or more language environments besides English:

- [Section C.1, "Process of Maintaining Translation Tables for Oracle BI"](#)
- [Section C.2, "About Translating Presentation Services Strings"](#)
- [Section C.3, "Changing the Default Currency in Analytics Applications"](#)

## C.1 Process of Maintaining Translation Tables for Oracle BI

The Oracle Business Intelligence Presentation layer supports multiple translations for any column name. When working with Oracle BI Answers or rendering a dashboard, users see their local language strings in their reports. For example, English-speaking and French-speaking users would see their local language strings in their reports. There are two kinds of application strings requiring translation in Oracle Business Intelligence:

- Metadata

Metadata strings are Analytics-created objects in the Oracle Business Intelligence repository such as Subject Area, Metrics, and Dimensions.

- Presentation Services

Presentation Services objects are end-user created objects such as Reports, Dashboards, and Pages. Translations for Presentation Services strings are stored in the captions.xml file. For more information on accessing these strings and changing the translations, see *Oracle Business Intelligence Presentation Services Administration Guide*.

This process includes the following tasks:

- [Section C.1.1, "Upgrading Oracle Business Intelligence Seed Data for Non-English Locales"](#)
- [Section C.1.2, "Externalizing Customer Metadata Strings"](#)
- [Section C.1.3, "Adding Custom Translations to the W\\_LOCALIZED\\_STRING\\_G Table"](#)

## C.1.1 Upgrading Oracle Business Intelligence Seed Data for Non-English Locales

If Oracle Business Intelligence data in your deployment is to be viewed in a language other than English, you must also import Locale seed data into a data warehouse table called `W_LOCALIZED_STRING_G`. This process must be performed once for each language the application users might select to run their web client.

During the Oracle Business Intelligence installation, a folder named `$INSTALLDIR\SeedData` was created, which contains a sub folder for each language. Within each language sub folder is a `.dat` file (the data to be imported) and an `.inp` file (the `WHERE` clause governing the import).

### C.1.1.1 Importing Locale Seed Data Into The Translation Table (`W_LOCALIZED_STRING_G`)

If the primary language being used is not English, you may have to import additional locale seed data (depending on the number of languages you use) as shown in the following procedures.

---

---

**Note:** This procedure can be performed only by a BI Administrator.

---

---

#### To verify creation of Translation Table (`W_LOCALIZED_STRING_G`) and corresponding indexes:

1. Verify that in the setting up of the Business Analytics Warehouse, tables included the creation of the `W_LOCALIZED_STRING_G` table (see page 105, Process of Creating Data Warehouse Tables).
2. Lookup the definitions of the indexes in the DAC and create them manually in the Business Analytics Warehouse. The names of the indexes are as follows:
  - `W_LOCAL_STRING_G_U1`
  - `W_LOCAL_STRING_G_P1`
  - `W_LOCAL_STRING_G_M1`
  - `W_LOCAL_STRING_G_M2`

**Note:** It is better to add these indexes to `W_LOCALIZED_STRING_G` prior to importing the locale seed data in the next section, in order to safeguard against inadvertently duplicating the data in the table.

#### To import Locale seed data into the Translation Table (`W_LOCALIZED_STRING_G`)

1. Open a command window and navigate to `$INSTALLDIR\SeedData\bin` folder.
2. Run the import command in step 3 after replacing these connection parameters with the values appropriate to your database environment:
  - `UserName`
  - `Password`
  - `ODBCDataSource`
  - `DatabaseOwner`
3. Run the import command:

```
$INSTALLDIR\SeedData\bin\dataimp /u $UserName /p $Password /c "$ODBCDataSource"
```

```
/d $DatabaseOwner /f $INSTALLDIR\SeedData\<>XX>\analytics_seed_<XXX>.dat /w y /q
100 /h Log /x f /i $INSTALLDIR\SeedData\<>XX>\metadata_upgrade_<XXX>_
<DBPlatform>.inp /l metadata_upgrade_<XXX>.log
```

---

**Note:** Replace the XX with the Oracle Business Intelligence two-letter language code (\_fr, \_it) and the XXX with the Siebel Systems three-letter code (FRA, ITA).

---

4. When you have finished importing the Locale seed data into the Translation Table (W\_LOCALIZED\_STRING\_G), configure the Initialization block in the Oracle BI Administration tool to connect to the database where this table resides.

---

**Note:** Unicode connectivity can be used to access databases that do not support Unicode.

---

## C.1.2 Externalizing Customer Metadata Strings

Metadata Strings are loaded by the Oracle BI Server from a database table. In the case of Oracle Business Intelligence applications, this table is W\_LOCALIZED\_STRING\_G in the data warehouse. The initialization block 'Externalize Metadata Strings' loads the strings for the Server. It is recommended that you run a test to make sure that this initialization block runs successfully. An example of the translation table is shown in [Table C-1](#).

**Table C-1** Example of W\_LOCALIZED\_STRING\_G Translation Table

MSG_NUM	MSG_TEXT	LANG_ID
CN_Customer_Satisfaction	Customer Satisfaction	ENU
CN_Customer_Satisfaction	Kundenzufriedenheit	DEU
CN_Customer_Satisfaction	Satisfação do cliente	PTB

By default, the Oracle Business Intelligence repository is configured to run in English only. To deploy in any other language, you must externalize the metadata strings, as described in the following procedure.

### To externalize metadata strings in the Oracle Business Intelligence repository

1. Stop the Oracle BI Server.
2. Using the Oracle BI Administration Tool in offline mode, open OracleBIAnalyticsApps.rpd.
3. Select the entire Presentation layer and right-click the mouse to display the menu.
  - From the pop-up menu, select Externalize Display Names. (A check mark appears next to this option the next time you right-click on the Presentation layer.)
  - Unselect the Presentation layer.

---

**Note:** When Externalize Display Names is checked, all metadata strings are read from the W\_LOCALIZED\_STRING\_G table in the data warehouse.

---

4. In the Physical layer, select the Externalized Metadata Strings database icon. Expand the tree.
5. Double-click Internal System Connection Pool.  
In the Connection Pool dialog General tab, the field Data source name should point to the data warehouse.
6. Click OK and exit the Oracle BI Administration Tool.
7. Restart the Oracle BI Server.

### C.1.3 Adding Custom Translations to the W\_LOCALIZED\_STRING\_G Table

When you add custom objects to the metadata and choose to externalize these objects (by right-clicking the object and checking the Externalize Display Name option), the Oracle BI Server looks for the translations (including those for the native language) in the W\_LOCALIZED\_STRING\_G table.

If you do not externalize the display names, you do not need to perform the following procedures.

---

---

**Note:** The custom Presentation layer objects show up only in the native language of the metadata (the language in which you added these new objects).

---

---

#### C.1.3.1 Adding String Translations for Analytics Metadata

The following procedure describes how to add string translations for Oracle Business Intelligence metadata to the W\_LOCALIZED\_STRING\_G table. This task occurs in any database administration tool, and in the Oracle BI Administration Tool.

##### To add string translations for Analytics metadata

1. Open a database administration tool and connect to your data warehouse database.
2. Query for the table named W\_LOCALIZED\_STRING\_G and add a new record to the table, as defined below in steps 4 to 8.
3. Obtain the Message Key from the Oracle BI Administration Tool as follows:
  - In the Oracle BI Administration Tool, right-click on the new Presentation layer metadata object and select Properties from the menu.
  - The Message key is displayed in the dialog under Custom Display Name. The Message key is the part that starts with CN\_.  
  
For example, double-click the Pipeline catalog folder in the Presentation layer. The Custom Display name is Valueof(NQ\_SESSION.CN\_Pipeline). CN\_Pipeline is the Message Key.
4. Enter your deployment language in the new record.
5. Enter the Message Type required (for example, Metadata, FINS\_Metadata).
6. Select the Message Level *AnalyticsNew*, then do the following:
  - In the Message Text column, add the translation of the object.
  - Check the flags (set to Yes) for the Translate and Active columns.
  - Set the Error Message # column to 0.

7. Enter the required Message Facility (for example, HMF, FIN).
8. Repeat Step 3 through Step 7 for each new metadata object string.
9. Exit the database administration tool, then restart the Oracle BI Server.

## C.2 About Translating Presentation Services Strings

The translations for such Presentation Services objects as report and page names are stored in the xxxCaptions.xml files available in OracleBIData\web\res\<language abbreviation>\Captions directories. In multiple language deployment mode, if you add any additional Presentation Services objects, such as reports and new dashboard pages, you also need to add the appropriate translations. Add these translations using the Catalog Manager tool. For more information on using this utility, see *Oracle Business Intelligence Presentation Services Administration Guide*.

## C.3 Changing the Default Currency in Analytics Applications

In Oracle Business Intelligence Applications, you may see a dollar sign used as the default symbol when amounts of money are displayed. In order to change this behavior, you must edit the currencies.xml file using the following procedure. The currencies.xml file is located in the following directories:

- Windows:  
`$INSTALLDIR\OracleBI\Web\config\`
- UNIX:  
`$INSTALLDIR/OracleBI/web/config`

### To change the default currency in Analytics Applications

1. In a text editor, open the currencies.xml file.
2. Look for the currency tag for the warehouse default (tag="int:wrhs"):

```
<Currency tag="int:wrhs" type="international" symbol="$" format="$#" digits="2"
displayMessage="kmsgCurrencySiebelWarehouse">
  <negative tag="minus" format="-$#" />
</Currency>
```

3. Replace the symbol, format, digits and negative information in the warehouse default with the information from the currency tag you want to use as the default.

For example, if you want the Japanese Yen to be the default, replace the contents of the warehouse default currency tag with the values from the Japanese currency tag (tag="loc:ja-JP"):

```
<Currency tag="loc:ja-JP" type="local" symbol="¥" locale="ja-JP" format="$#"
digits="0">
  <negative tag="minus" format="-$#" />
</Currency>
```

When you are finished, the default warehouse currency tag for Japanese should look like the following example:

```
<Currency tag="int:wrhs" type="international" symbol="¥" format="$#" digits="0"
displayMessage="kmsgCurrencySiebelWarehouse">
  <negative tag="minus" format="-$#" />
</Currency>
```

4. Save and close the currencies.xml file.

---

---

# Integrating Interactive Dashboards and Operational Applications Data

This appendix describes the additional configuration steps required for you to run an Oracle Business Intelligence application with one of Siebel (CRM) and Oracle E-Business Suite (operational) applications.

**Tip:** See Oracle Business Intelligence Server Administration Guide before performing any of the tasks in this section.

The integration of Oracle Business Intelligence with a Siebel operational application involves two general processes:

- [Section D.1, "Importing Oracle's Siebel Industry Applications Seed Data"](#)
- [Section D.2, "Completing the Initialization in the Siebel Operational Application"](#)

## D.1 Importing Oracle's Siebel Industry Applications Seed Data

Oracle Business Intelligence seed data is not installed with Oracle's Siebel Industry Applications. You must import the seed data into your database after the Oracle Business Intelligence installation is completed.

### To import Analytics seed data into a transactional database

1. Obtain the required language.inp and .dat files from the Oracle Business Intelligence language folder `\OracleBI\SeedData\bin\l_xx`, where `xx` is the two-letter code for the language you want to import.
2. Copy the .dat and corresponding .inp file from the language folder to the server installation `\bin` folder.
3. From the command prompt in `\OracleBI\SeedData\bin`, run the following command:

```
dataimp /u $USERNAME /p $PASSWORD /c "$ODBCDataSource" /d $Tableowner /f analytics_seed_<XXX>.dat /i metadata_upgrade_<XXX>_<DBPlatform>.inp /w y
```

Replace the `XXX` with the three-letter code (FRA, ITA) and the `DBPlatform` with the abbreviation for the database platform being used. For example:

```
dataimp /u sadmin /p sadmin /c JPN_CRMDEV1 /d siebel /f analytics_seed_JPN.dat /i metadata_upgrade_JPN_db2.inp /w y
```

For information about merging content into Oracle Business Intelligence Presentation Services, see the topics about using the Catalog Manager in *Oracle Business Intelligence Presentation Services Administration Guide*.

## D.2 Completing the Initialization in the Siebel Operational Application

Once you have configured your Oracle BI Server and are able to access the dashboards, you need to update the Siebel operational application to view Analytics dashboards from within the Siebel operational application. Completing the initialization in the Siebel operational application involves the following processes:

- Changing the operational application host name to the host name of the machine that runs Oracle Business Intelligence Presentation Services. See the following topics:
  - [Section D.2.1, "Updating the Siebel Operational Application"](#)
  - [Section D.2.2, "Reapplying Customized Style Sheets"](#)
  - [Section D.2.3, "How to Configure Oracle Business Intelligence with Oracle's Siebel Web Extension \(SWE\)"](#)
    - \* [Section D.2.4, "Creating a Virtual IP Address for the SWE and Oracle BI Presentation Services"](#)
    - \* [Section D.2.5, "Testing the Virtual IP Configuration for Oracle Business Intelligence and the SWE"](#)
  - [Section D.2.6, "How to Configure Oracle BI Action Links"](#)
    - \* [Section D.2.7, "Testing Action Links"](#)
- Customizing the operational application home page, dashboards, or content, or add views to a dashboard. See the following topics:
  - [Section D.2.8, "Process of Integrating Analytics and Oracle's Siebel Operational Applications"](#)
    - \* [Section D.2.9, "Accessing Optional Analytics Applications"](#)
    - \* [Section D.2.10, "Configuring the Browser Settings for Operational Applications"](#)
  - [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#)
    - \* [Section D.2.12, "Customizing Oracle BI Content on the Siebel Operational Application Home Page"](#)
    - \* [Section D.2.13, "Determining the Oracle BI Report Path Argument"](#)
    - \* [Section D.2.14, "Adding Views for Custom Oracle BI Interactive Dashboards"](#)
    - \* [Section D.2.15, "Verifying NQHOST Setup for Pharma Disconnected Analytics Client"](#)
    - \* [Section D.2.16, "Integrating the Pharma Application Home Page with the Pharma Disconnected Analytics Client"](#)
    - \* [Section D.2.17, "Importing Pharma Application Target Lists to the Pharma Disconnected Analytics Client"](#)

### D.2.1 Updating the Siebel Operational Application

The following task changes the Siebel operational application host name to the host name of the machine that runs Oracle Business Intelligence Presentation Services.

#### To update the Siebel operational application



1. Open your Siebel operational application and login as SADMIN.
2. Navigate to View, then Site Map.
3. Click Integration Administration screen.
4. Click Host Administration view.
5. Query for NQHOST in the Virtual Name column:
  - In Windows, change the host name from `<AnalyticsServerName>` to the host name of the machine that runs Oracle BI Presentation Services.
  - In AIX or Solaris, change the host name from `<AnalyticsServerName>` to the name of the port.

For example, `servername.siebel.com:8080`

---

**Note:** For UNIX platforms only: You may need to add the domain name suffix to the server name in order to make sure that action links work on the Oracle Business Intelligence user interface.

---

6. Log out of the application and log back in.

## D.2.2 Reapplying Customized Style Sheets

For Oracle's Siebel Business Analytics versions 7.7 and later, new styles have been appended to the following style sheets:

- Go.css
- PortalBanner.css
- PortalContent.css
- Views.css

These new classes are identified in the style sheets. In this version of Oracle Business Intelligence, new styles and files must be added to the underlying style sheet (for example, to `s_Siebel7`). For complete functionality, any custom styles require similar updating. In addition, `views2.css` and some other files have been added to the `s_` directory. For custom column formatting to work properly, references to font sizes and families should also be removed from the `td` styles in `PortalBanner.css`, `PortalContent.css`, `Views.css`.

Because of these changes to styles, review your deployment's customizations manually, reapply them, and test them thoroughly to ensure that there are no problems.

After you have tested them, but before you copy the default `views.css` files back to the implementation server, perform the following task on the server to clear the server caches.

### To clear the server caches and restore your default views

1. Shut down Oracle Business Intelligence Server, Oracle BI Presentation Service and IIS.
2. Remove your custom `views.css` from the directory where it has been installed.

For example:

```
$INSTALL\Web\App\Res\s_Siebel7\b_mozilla_4
```

or

`$INSTALL\OracleBIData\Web.`

3. Clear the Oracle BI Presentation Services Server Cache.

In the `C:\WINNT\Temp` directory, delete the `nQs_*.temp` files.

4. Clear the Browser Cache.

From the Internet Explorer menu, navigate to Tools, then Internet Options, then Settings, then View Files, and delete all the files in these folder.

5. Restore the default `views.css` files to the appropriate folder.
6. Restart the Analytics Server, Oracle BI Presentation Services and IIS.

## D.2.3 How to Configure Oracle Business Intelligence with Oracle's Siebel Web Extension (SWE)

Whenever you run Oracle Business Intelligence and Siebel Web Extension (SWE) on separate machines, you must perform additional configuration steps in order for action links and interactive charts to work. If, for example, you plan to run the SWE and Oracle Business Intelligence Presentation Services on different Web servers, you must use some kind of networking or load balancing mechanism to create a single logical domain (or virtual IP address) for the two machines.

---

---

**Caution:** When Oracle BI Applications Server and Oracle BI Presentation Services are installed on different machines and load balancing software is used to create a single virtual IP address, the action links feature works only if you log onto the Siebel application using a virtual Web browser. Action links fail on a dedicated client if the SWE and Oracle BI Presentation Services servers are on separate machines. Either use a virtual Web browser, or install the Oracle BI Applications Web Server and Oracle BI Presentation Services on the same machine.

---

---

When one virtual IP address is created for two machines, the Web browser accesses one IP address and is still routed to different physical machines, based on the port accessed. From the browser, it appears that both servers are running on the same IP address.

You can use any of several physical methods to create a single logical domain, such as running SWE and Oracle Business Intelligence Presentation Services on a single machine if you are not load balancing the SWE, or using a router to do the mapping, or using load balancing software. Your company must determine the best mechanism to accomplish this routing given the topology being used.

Configuring Oracle Business Intelligence to work with Siebel Web Extension includes the following tasks:

- [Section D.2.4, "Creating a Virtual IP Address for the SWE and Oracle BI Presentation Services"](#)
- [Section D.2.5, "Testing the Virtual IP Configuration for Oracle Business Intelligence and the SWE"](#)

## D.2.4 Creating a Virtual IP Address for the SWE and Oracle BI Presentation Services

This task is part of the [Section D.2.3, "How to Configure Oracle Business Intelligence with Oracle's Siebel Web Extension \(SWE\)"](#).

You create a virtual IP address for the Siebel Web Engine (SWE) and Oracle BI Presentation Services in order to make it appear that all servers are running on the same virtual machine. The easiest way to do this is to configure Oracle BI Presentation Services to run on a different port (for example, port 84) from SWE (which usually runs on port 80).

For example, SWE is load-balanced across *<machine1>*:port 80 and *<machine2>*: port 80, and Oracle BI Presentation Services is running on *<machine3>*:port 84, and the virtual address is defined as `http://siebel.company.com`. Therefore, the network or load-balancing software should be configured to route requests like `http://siebel.company.com` to *<machine1>* and *<machine2>*, and to route requests like `http://siebel.company.com:84` to *<machine3>*.

### To create a virtual IP address for Siebel Web Engine and Oracle BI Presentation Services

1. On the network, set up CSS to direct requests from *<virtual domain>* to *<physical Siebel Web Server Extension machine>*:
  - Where the acronym CSS represents the load-balancer or router used to do the virtual IP configuration.
  - Where *<virtual domain>* is the virtual IP prefix that users enter to navigate to the Siebel applications (in the preceding example, this is `http://siebel.company.com`).
2. On the network, set up CSS to direct requests from *<virtual domain>*:84 to *<physical Oracle BI Presentation Services machine>*:84.
3. In the Siebel application, using the Siebel Administration screen, set the NQHost parameters for Oracle Business Intelligence Symbolic URLs to point to the *<virtual domain>*:84, instead of directly to the Oracle Business Intelligence Presentation Services server physical machine.

---

**Note:** If you are running Oracle's Siebel Business Analytics 7.5.3 instead of version 7.7 or later, perform the following additional step.

---

4. In the Siebel application on the Oracle Business Intelligence Presentation Services server machine, locate the registry setting `\SOFTWARE\Siebel Systems, Inc.\Siebel Analytics\Web\7.5\Charts`.
5. Add a new key, `ForceFileBasedPainter`, and enter TRUE into the Data string.

## D.2.5 Testing the Virtual IP Configuration for Oracle Business Intelligence and the SWE

This task is part of the [Section D.2.3, "How to Configure Oracle Business Intelligence with Oracle's Siebel Web Extension \(SWE\)"](#).

Use a client browser to verify that Oracle Business Intelligence and SWE work when accessed directly through a physical machine address, using the following procedure. For *<virtualdomain>*, substitute the Virtual IP address you created in [Section D.2.4, "Creating a Virtual IP Address for the SWE and Oracle BI Presentation Services"](#).

### To test the Virtual IP configuration with Oracle Business Intelligence and SWE

1. In a client browser, type `<virtual domain>:84/analytics`. The Oracle Business Intelligence logon appears.
2. In a client browser, type `<virtual domain>/callcenter` (or other Siebel application). The SWE appears.
3. Navigate to an Analytics screen within the Siebel application to see if Oracle Business Intelligence appears.
4. Interact with Oracle Business Intelligence charts and action links.

## D.2.6 How to Configure Oracle BI Action Links

Oracle Business Intelligence applications contain prebuilt action links in reports and dashboards. Action links are a way to navigate from a user's analytics dashboard to a record in a Siebel operational application, such as Oracle's Siebel Call Center.

For example, you can drill down directly from a Order Management Analytics dashboard to a specific record in a Sales view. The link is based on the row-ID column contained in a report. You can also create new action links for any existing report. For how to create Interactive Oracle Dashboards action links, see *Oracle Business Intelligence Presentation Services Administration Guide*.

Depending on your deployment, you may need to perform additional configuration of your Oracle Business Intelligence application in order for the action links feature to work properly. The following tasks are relevant to the configuration of your application for action links:

- [Section D.2.7, "Testing Action Links"](#)

## D.2.7 Testing Action Links

This task is part of the [Section D.2.6, "How to Configure Oracle BI Action Links"](#).

Use the following procedure to make sure that action links you create work correctly.

### To test an Oracle Business Intelligence action link

1. Log on using your system.  
Go to Service Analytics screen, then Employee Analytics, then To Do List.
2. Click the action link on the pie chart in Overdue Activities.  
You should be in Overdue Activities Detail in Analytics.
3. Click any ROW\_ID action link.  
You should be in the Employee screen in Oracle's Siebel Call Center operational application.

## D.2.8 Process of Integrating Analytics and Oracle's Siebel Operational Applications

The following tasks are described for the process of integrating your Oracle Business Intelligence application with your Siebel operational application:

- [Section D.2.9, "Accessing Optional Analytics Applications"](#)
- [Section D.2.10, "Configuring the Browser Settings for Operational Applications"](#)
- [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#)

## D.2.9 Accessing Optional Analytics Applications

This task is part of [Section D.2.8, "Process of Integrating Analytics and Oracle's Siebel Operational Applications"](#)

Depending on the options you purchased with your Siebel operational application, you must perform additional steps in order to access the corresponding Oracle Business Intelligence options. [Table D-1](#) shows the additional options for Oracle Business Intelligence.

**Table D-1 Oracle Business Intelligence Applications Options**

Siebel Application	Options
Sales	Order Management Analytics, Usage Accelerator option.
Service	Service Analytics, Contact Center Telephony Analytics option.

By default, the dashboards and reports contained in these optional areas are hidden. If, for example, you purchased Order Management Analytics with your Sales application, you must perform the additional steps shown in the following procedure to access the Sales Analytics.

### To turn on options for Sales Analytics

1. Log in to Oracle Business Intelligence as Administrator.
2. Navigate to Answers, then Oracle BI Presentation Services Administration and select the option to manage Presentation Services groups and users.
3. Locate the Web Group corresponding to your option.

The Web Group options are shown in the following list.

- No Forecasting
  - No Forecasting Lite
  - No Universal Queuing
  - No Email Response
  - No Service Agreements
  - No Partner Marketing
  - No Partner ERM
  - No Partner ISS
4. Click on the Edit icon.
  5. Under the Group Membership section, click the delete icon (X) to delete Analytics Users from this group.
  6. Click Finished and log out of the application.
  7. Log in again to access the additional optional dashboards and reports.

## D.2.10 Configuring the Browser Settings for Operational Applications

---

**Note:** This topic applies only to Oracle's Siebel Business Analytics version 7.7.1 and Siebel operational applications version 7.7.

---

This task is part of [Section D.2.8, "Process of Integrating Analytics and Oracle's Siebel Operational Applications"](#).

If the Siebel Server and the Oracle Business Intelligence Server URLs are from different domains, you may need to change the browser settings for your Oracle Business Intelligence application. Perform the following procedure.

---

---

**Note:** You do not need to change browser settings if the Siebel Server and the Oracle Business Intelligence Server URLs are from the same domain.

---

---

#### **To change browser settings for Oracle Business Intelligence operational applications**

1. In your browser, navigate to Tools, then Internet options, then Security tab.
2. Select Internet zone and click Custom Level.
3. In the Miscellaneous > Access data sources across domains setting, select the Enable radio button and click OK.
4. Select Local Intranet zone and click Custom Level.
5. In the Miscellaneous > Access data sources across domains setting, select the Enable radio button and click OK.
6. Click OK to exit browser Tools.

### **D.2.11 Process of Customizing Oracle BI Application Home Page and Dashboards**

The process of customizing your Oracle Business Intelligence application's home page and dashboards may include the following tasks:

- [Section D.2.12, "Customizing Oracle BI Content on the Siebel Operational Application Home Page"](#)
- [Section D.2.13, "Determining the Oracle BI Report Path Argument"](#)
- [Section D.2.14, "Adding Views for Custom Oracle BI Interactive Dashboards"](#)
- [Section D.2.15, "Verifying NQHOST Setup for Pharma Disconnected Analytics Client"](#)
- [Section D.2.16, "Integrating the Pharma Application Home Page with the Pharma Disconnected Analytics Client"](#)
- [Section D.2.17, "Importing Pharma Application Target Lists to the Pharma Disconnected Analytics Client"](#)

For a list of Pharma Disconnected Analytics components that are installed with Oracle BI Applications, see [Appendix G, "Pharma Disconnected Analytics Administration - Supplemental Information"](#).

### **D.2.12 Customizing Oracle BI Content on the Siebel Operational Application Home Page**

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#).

Oracle Business Intelligence applications are integrated with Siebel operational applications using the symbolic URL infrastructure. The following task describes how

to use symbolic URLs to link a new report to a Siebel operational application home page and how to add new Analytics Dashboards to the Siebel operational application. The symbolic URL specifies how the HTTP request to the external application should be constructed and to defines any arguments and values to be sent as part of the request.

---

**Note:** For more information on the symbolic URL infrastructure, see Siebel Portal Framework Guide. This topic assumes that you have successfully built a symbolic URL to link some external content, using the instructions given in Siebel Portal Framework Guide.

---

For each Analytical report on a Siebel operational application home page, there is a symbolic URL record defined that links the home page to the Analytics report. If you have not already done so, you need to create a new home page and set it up to use a symbolic URL.

---

**Note:** For Oracle Business Intelligence Disconnected settings, see [Section D.2.15, "Verifying NQHOST Setup for Pharma Disconnected Analytics Client"](#).

---

The figure below shows example Inline and IFrame symbolic URL arguments.

**Figure D-1 Examples of Symbolic URL Arguments.**

Inline

Name	Required Argument	Argument Type	Argument Value	Append as Argument	Substitute in Text	Sequence #
Cmd	✓	Constant	Go	✓		1
Path	✓	Constant	/shared/Service/Service Requests	✓		2
Syndicate	✓	Constant	Siebel	✓		3
nqUser	✓	Command	UseSiebelLoginId	✓		4
nqPassword	✓	Command	UseSiebelLoginPassword	✓		5

IFrame

Name	Required Argument	Argument Type	Argument Value	Append as Argument	Substitute in Text	Sequence #
IFrameLogin:Cmd	✓	Constant	Logon	✓		1
Cmd	✓	Constant	PortalPages	✓		2
IFrameLogin:nqUser	✓	Command	UseSiebelLoginId	✓		3
IFrameLogin:nqPassword	✓	Command	UseSiebelLoginPassword	✓		4
PortalPath	✓	Constant	/shared/Service/_Portal/Activities	✓		5
PostRequest	✓	Command	PostRequest	✓		6
IFrameLogin:Syndicate	✓	Constant	Siebel	✓		7

**To configure a new Home Page to use a symbolic URL**

1. Make sure the symbolic URL has been set up.
2. Launch the Siebel operational application and navigate to the Integration Administration, then Symbolic URL Administration view.
3. Query for the symbolic URL that has been set up for Analytics.

The name of this symbolic URL should be exactly the same as the calculated value of the field that was added to the Business Component. For example, you may have a symbolic URL named HomePageAnalytics.

- In the URL field, enter the Web URL. For example:

`http://NQHOST/Analytics/saw.dll`

For the Host Name, choose the Analytics Server Name from the drop-down list. The following table shows the other parameters for reports.

Parameter	Value
NQHOST	A virtual name in the URL that points to the Oracle BI Presentation Services machine
Fixup Name	Inside Applet
SSO Disposition	Inline
Oracle BI Presentation Services application name	Select from the drop-down list

- Create the appropriate symbolic URL Arguments.

These arguments depend upon the particular Analytics report that you are trying to display. The Argument Values should be the same for any Analytics report you work with, except for the Path Argument Value. Use the procedure in [Section D.2.13, "Determining the Oracle BI Report Path Argument"](#) to determine the path to the Analytics report.

### D.2.13 Determining the Oracle BI Report Path Argument

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#).

The Path argument tells the symbolic URL the path to the report on the Oracle BI Presentation Services. (For example, `/shared/Sales/Pipeline/Overview/Top 10 Deals`.) Use the following procedure to determine the path to the Analytics report.

#### To determine the path to the report

- Log on to your Oracle BI Presentation Services as an Administrator.
- In the Siebel operational application, navigate to Answers, then Oracle BI Presentation Services Administration.
- Select Manage Analytics Catalog, and then navigate to your report.
- Add this path name to the Symbolic URL argument.

The following table shows the symbolic URL path arguments for reports.

Name	Type	Path Argument Value	Append	Sequence #
Cmd	Constant	GO	Y	1
Path	Constant	<code>/shared/Sales/Pipeline/Overview/Top 10 Deals</code>	Y	2
Syndicate	Constant	Siebel	Y	3
nQUser	Command	UseSiebelLoginId	Y	4
nQPassword	Command	UseSiebelLoginPassword	Y	5
PostRequest	Command	PostRequest	Y	6



## D.2.14 Adding Views for Custom Oracle BI Interactive Dashboards

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#) and is similar to that of adding Oracle Business Intelligence reports to the home page:

- Using Oracle's Siebel Tools, set up a new view.
- In the Siebel operational application, define a symbolic URL for that view.

For more information on how to set up a view to use Symbolic URLs, see Siebel Portal Framework Guide.

### To configure the symbolic URL for Analytics dashboards

1. Define a Symbolic URL.
  - a. Navigate to Site Map, then Integration Administration, then Symbolic URL Administration.
  - b. In the Symbolic URL Administration view, add a new record.
2. Define Symbolic URL arguments.
  - a. Navigate to Symbolic URL Administration.
  - b. In the Symbolic URL Administration list, select the Symbolic URL you want to configure.

## D.2.15 Verifying NQHOST Setup for Pharma Disconnected Analytics Client

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#).

Pharma Disconnected Analytics is a prebuilt Disconnected Analytics application for Oracle's Siebel Pharma Sales. When you run Oracle Business Intelligence Disconnected Client, your client application runs against a local instance of the database on your machine. You need to confirm that NQHOST is using the correct port number and DLL files to work with the Pharma Disconnected Analytics Client.

### To verify the NQHOST setup

1. Run the Disconnected Client application against your local database.
2. Log in as Administrator.

You must have access to the following Integration Administration Screen views:

- WI Host Admin View
- WI Symbolic URL Admin View

3. Navigate to Administration – Integration.
4. Click WI Symbolic URL List, and search for NQHOST in Virtual Name field.

The Host Name value should be localhost:9762.

5. Select the Host Administration View from the drop-down list and choose Symbolic URL Administration.
6. The following table shows Disconnected Client symbolic URLs to search for.

For each of the URLs, the URL field is:

```
http://NQHOST/Analytics/sawd.dll
```

The Disconnected Client Symbolic URL Names are:

- ePharmaHomePageAnalytics1
- ePharmaHomePageAnalytics2
- ePharmaHomePageAnalytics3
- ePharmaHomePageAnalytics4
- SISMLSDistrictManagerDashboard1
- SISMLSSalesRepDashboard1
- SiebelAnswers
- SiebelDelivers

The following table shows the symbolic URL path arguments for Oracle BI Answers and Oracle BI Delivers.

Name	Type	Path Argument Value	Append	Sequence #
Cmd	Constant	Answers for SiebelAnswers	Y	1
		Delivers for SiebelDelivers		
nQUser	Command	UseSiebelLoginId	Y	2
nQPassword	Command	UseSiebelLoginPassword	Y	3

## D.2.16 Integrating the Pharma Application Home Page with the Pharma Disconnected Analytics Client

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#).

If you have licensed both the Siebel Pharma operational application and Oracle Business Intelligence Disconnected Client, you must change part of the operational application configuration file in order to access Analytics reports on the LS Analytics Home Page and to allow Create Target List integration.

### To allow integration of Pharma Analytics Home Page with Pharma Disconnected Analytics

1. On the machine where you have installed the Siebel operational application, navigate to `$INSTALLDIR\sea77\siebsrvr\BIN\ENU\epharma.cfg`.
2. Using a text editor, open the file `epharma.cfg`.
3. In the Local section, find the parameter `UseCachedExternalContent`.

The default value for this parameter is TRUE.

---

**Note:** If this parameter remains set to TRUE, Analytics reports on the Home Page return an error.

---

4. Set the `UseCachedExternalContent` parameter to FALSE.
5. Save and close the file.

## D.2.17 Importing Pharma Application Target Lists to the Pharma Disconnected Analytics Client

This task is part of the [Section D.2.11, "Process of Customizing Oracle BI Application Home Page and Dashboards"](#).

If you have licensed both the Siebel Pharma operational application and Oracle Business Intelligence Disconnected Client, you must change part of the operational application configuration file in order to allow you to import the contacts from the Pharma Analytics Create Target List function into the Disconnected Analytics Client.

### To allow importation of Target List contacts into Oracle Business Intelligence Disconnected Client

1. On the machine where you have installed the Siebel operational application, navigate to `$INSTALLDIR\sea77\siebsrvr\BIN\ENU\epharma.cfg`.
2. Using a text editor, open the file `epharma.cfg`.
3. In the Siebel Client section, find the parameter `EnableFQDN`.

The default value for this parameter is `TRUE`.

---

**Note:** If this parameter remains set to `TRUE`, the Create Target List functionality is disabled in the Pharma Disconnected Analytics Client.

---

4. Set the `EnableFQDN` parameter to `FALSE`.
5. Save and close the file.
  - a. In the Symbolic URL Arguments list, enter the arguments that need to be sent to the external host.

The following table shows the symbolic URL argument definitions for dashboards.

Parameter	Value
Fixup Name	Default
SSO Disposition	IFrame
Oracle BI Presentation Services application name	Select from the drop-down list.

The following table shows the symbolic URL path arguments for dashboards.

Name	Type	Value	Append	Sequence #
Cmd	Constant	Dashboard	Y	1
PortalPath	Constant	/	Y	2
Page	Constant	Sales	Y	3
Syndicate	Constant	Siebel	Y	4
nQPassword	Command	UseSiebelLoginPassword	Y	5
nQUser	Command	UseSiebelLoginId	Y	6

6. In Oracle Business Intelligence, create the dashboards.

---

---

**Note:** Only an Oracle BI Presentation Services administrator can perform this step.

---

---

7. In Oracle Business Intelligence, set up the dashboards to your requirements.

## D.3 Viewing Information About Preconfigured Dashboards

If your organization has prebuilt applications installed, you can use Catalog Manager to locate and view information about preconfigured dashboards. You may first have to expose the dashboards and requests.

### D.3.1 Exposing Dashboards and Requests

Depending on the Oracle Business Intelligence options your organization purchased, you may need to expose these options before the associated dashboards and requests can be viewed in Oracle BI Presentation Services and in Catalog Manager. This applies to sites that have the following prebuilt applications and options:

- Oracle Order Management Analytics, with the Forecasting and Incentive Compensation options.
- Oracle Service Analytics, with the Universal Queuing, Email Response, and Agreements options.
- Oracle Partner Analytics, with the Partner Marketing, Partner ERM, and Partner ISS options.

These options need to be exposed using the administration feature in Oracle BI Presentation Services.

### D.3.2 Locating Dashboard and Page Names

In Catalog Manager, the Presentation Catalog distributed with prebuilt applications has the following structure:

Presentation Catalog > shared folder > prebuilt application name > \_portal folder > dashboard name > dashboard page name

---

---

# Configuring Metadata for Analytics Applications

Oracle's Siebel applications are shipped with certain Oracle Business Intelligence metadata. These metadata repositories are included with Oracle's Siebel operational applications and Oracle's Siebel Industry Applications. All the rules for security, data modeling, aggregate navigation, caching, and connectivity for the Oracle Business Analytics Warehouse are stored in metadata repositories on the Oracle Business Intelligence Server.

This appendix describes configuration necessary for the Oracle Business Intelligence metadata for Oracle's Siebel operational applications, specifically those areas that a Siebel administrator may need to adjust. These administrative tasks generally cover two subareas:

- Dashboard content of Oracle's Siebel operational applications
- Metadata requirements for Oracle's Siebel operational applications

This appendix contains the following topics:

- [Section E.1, "What Are Oracle's Siebel Operational Applications?"](#)
- [Section E.2, "Updating Financial Analytics Logical Table Sources"](#)
- [Section E.3, "Externalizing Financial Services Metadata Translation Strings"](#)
- [Section E.4, "Disabling the Financial Analytics Logical Table Sources"](#)
- [Section E.5, "Developing and Deploying Predictive Scores"](#)
- [Section E.6, "Analytics Metadata Requirements for Oracle's Siebel Industry Applications"](#)

## E.1 What Are Oracle's Siebel Operational Applications?

Oracle's Siebel Operational Applications are built around a general business function, such as Oracle's Siebel Sales, Oracle's Siebel Service, and Oracle's Siebel Call Center.

Oracle's Siebel Industry Applications are built around specific industries' business practices:

- Financial Services
- Consumer Sector
- Communications, Media, and Energy (CME)
- Pharma Sales

---

---

**Note:** You can license one or more Siebel Industry Applications for Oracle Business Intelligence. Licensing allows access to particular dashboards, subject areas, and reports.

---

---

## E.2 Updating Financial Analytics Logical Table Sources

In the Oracle BI repository file, logical table sources are set by default to settings in Oracle's Financial Analytics family of products (Finance Sales Analytics, Finance Service Analytics, Finance Marketing Analytics, Finance Institutional Analytics, Finance Retail Analytics), the Healthcare Analytics family of products (Oracle Healthcare Sales Analytics, Oracle Healthcare Service Analytics, Oracle Healthcare Marketing Analytics, Oracle Healthcare Partner Manager Analytics), and the Oracle Insurance Analytics family of products (Oracle Insurance Partner Manager Analytics, Oracle Insurance Sales Analytics, Oracle Insurance Service Analytics, Oracle Insurance Marketing Analytics, Oracle Insurance Partner Manager Analytics). Before using these Financial Services Analytics applications, you must update the logical sources for two tables in the Oracle Business Intelligence repository file. These logical sources must be deactivated in order for Oracle's Financial Analytics family of products (Finance Sales Analytics, Finance Service Analytics, Finance Marketing Analytics, Finance Institutional Analytics, Finance Retail Analytics), the Healthcare Analytics family of products (Oracle Healthcare Sales Analytics, Oracle Healthcare Service Analytics, Oracle Healthcare Marketing Analytics, Oracle Healthcare Partner Manager Analytics), and one of the Oracle Insurance Analytics family of products (Oracle Insurance Partner Manager Analytics, Oracle Insurance Sales Analytics, Oracle Insurance Service Analytics, Oracle Insurance Marketing Analytics, Oracle Insurance Partner Manager Analytics) reports to point to the correct logical model and retrieve the correct data. Do this by deactivating the sources for the Fact-Asset logical table in the Core subject area and activating the FINS logical sources, as shown in the following procedure.

1. Using the Oracle BI Administration Tool, open the Analytics Repository (OracleBIAnalyticsApps.rpd).
2. Go to the Business Model and Mapping dialog (the logical layer dialog) and open the Core folder.
3. Scroll down to the Fact - CRM - Asset logical table and open its Sources folder.
4. In the list of logical table sources, right-click Fact\_ASSET\_F\_FINS.
5. Select Properties.
6. Click the General tab in the Properties dialog and make sure that the Active check box is checked. If it is not, check it.
7. In the list of logical table sources, right-click W\_ASSET\_F.
8. Select Properties.
9. Click the General tab in the Properties dialog and make sure that the Active check box is deselected.
10. Click OK and save the repository.
11. Restart Oracle BI Server.

## E.3 Externalizing Financial Services Metadata Translation Strings

The Financial Services applications use a different set of translation strings from other Siebel operational applications.

You must externalize the metadata strings in the Analytics repository.

### To externalize metadata strings in the Analytics repository

1. Using the Oracle BI Administration Tool, open the OracleBIAnalyticsApps.rpd.
2. Select the entire presentation layer and right-click the mouse to display the menu.
3. Choose Display Names.

With this configuration, all metadata strings are read from an external Siebel operational application database, specifically from the table W\_LOCALIZED\_STRING\_G.

4. Make sure that the connection pool Externalized Metadata Strings points to the Siebel operational application database and is working correctly.
5. Go to the Manage Variables menu and locate the initialization block External Metadata Strings.

Double-click on the initialization block to open the Edit dialog.

6. In the Initialization SQL area, change the SQL:

from

```
select MSG_NUM, MSG_TEXT
from VALUEOF(OLAP_TBO).W_LOCALIZED_STRING_G
where MSG_TYPE = 'Metadata' and
LANG_ID = decode('VALUEOF(NQ_SESSION.WEBLANGUAGE)'... [more]
```

to

```
select MSG_NUM, MSG_TEXT
from VALUEOF(OLAP_TBO).W_LOCALIZED_STRING_G
where MSG_TYPE = 'FINS_Metadata' and
LANG_ID = decode('VALUEOF(NQ_SESSION.WEBLANGUAGE)'... [more]
```

7. Click OK to commit the changes and save the repository.
8. Restart the Oracle BI Server.

## E.4 Disabling the Financial Analytics Logical Table Sources

In the Analytics Repository file, logical table sources are set by default to settings in Oracle's Siebel Financial Analytics, the Healthcare Analytics family of products (Oracle Healthcare Sales Analytics, Oracle Healthcare Service Analytics, Oracle Healthcare Marketing Analytics, Oracle Healthcare Partner Manager Analytics), and the Oracle Insurance Analytics family of products (Oracle Insurance Partner Manager Analytics, Oracle Insurance Sales Analytics, Oracle Insurance Service Analytics, Oracle Insurance Marketing Analytics, Oracle Insurance Partner Manager Analytics). If you are using any *Siebel Industry application that is not* Oracle's Siebel Financial Services Analytics, you must first update the logical table sources in the Analytics Repository file.

These logical table sources must be deactivated in order for your Siebel Industry application Analytics reports to point to the correct logical model and retrieve the correct data. Do this by deactivating the FINS sources for the logical tables in the Core

subject area and activating the other logical sources, as shown in the following procedure.

---

---

**Note:** Before performing the following procedure, shut down the Oracle BI Server.

---

---

**To update Logical Table sources for Oracle's Siebel Industry Applications Analytics**

1. Using the Oracle BI Administration Tool, open the Analytics Repository (OracleBIAnalyticsApps.rpd).
2. Go to Business Model and Mapping dialog (the logical layer dialog) and open the Core folder.
3. Scroll down to the Fact - CMR - Asset logical table and open its Sources folder.
4. In the list of logical table sources, right-click Fact\_W\_ASSET\_F\_FINS.  
Select Properties.
5. Click the General tab in the Properties dialog and deselect the Active check box.
6. In the list of logical table sources, right-click W\_ASSET\_F.  
Select Properties.
7. Click the General tab in the Properties dialog.  
Make sure that the Active check box is checked. If it is not, check it.  
Click OK and save the repository.
8. Restart Oracle BI Server.

## E.5 Developing and Deploying Predictive Scores

The Loyalty Management Dashboard and several Oracle Business Intelligence subject areas use customer scores generated from Oracle Real-Time Decisions. Oracle Real-Time Decisions uses mathematical models to predict customer behavior. For customer scoring to be made available for analysis in Oracle Business Intelligence, CME metadata is provided which maps these customer scores to dashboards and subject areas.

The following procedure describes the process of developing and deploying these predictive scores.

**To develop and deploy predictive scores**

1. Generate predictive scores using Oracle Real-Time Decisions.

---

---

**Note:** This is performed outside of the Siebel operational application.

---

---

2. Integrate the scores into the Oracle Business Analytics Warehouse.  
Once this is completed, scores may be viewed in the Siebel operational application by accessing the Accounts, then Profiles, then Loyalty Profile view.
3. Load the integrated scores into the Oracle Business Analytics Warehouse during the extraction, transformation, and loading (ETL) process.



4. After the scores are loaded into the Oracle Business Analytics Warehouse, map them to the following Oracle Business Intelligence metadata fields:
  - Churn Score
  - Customer Lifetime Value Score
  - Upsell Score
  - Cross-Sell Score
  - Financial Risk Score

In conjunction with other associated metadata, these fields are primarily used to populate the Loyalty Management dashboard.

## E.6 Analytics Metadata Requirements for Oracle's Siebel Industry Applications

Some metadata needs to be set up properly in the Oracle Business Analytics Warehouse for it to be displayed accurately in Oracle Business Intelligence. The following topics describe the metadata structure for each of the following Oracle's Siebel Industry Applications:

- [Section E.6.1, "Oracle's Siebel Consumer Sector Dashboards and Pages"](#)
- [Section E.6.2, "Oracle's Siebel Consumer Sector Data Requirements"](#)
- [Section E.6.3, "Oracle's CME family of products \(Oracle Communications, Media and Energy Sales Analytics\)"](#)
- [Section E.6.4, "Oracle's Siebel CME Dashboards and Pages"](#)
- [Section E.6.5, "Oracle's Siebel CME Data Requirements"](#)
- [Section E.6.6, "Oracle Pharma Sales Analytics Dimensions"](#)
- [Section E.6.7, "Dimensions Specific to Subject Areas in Oracle Pharma Sales Analytics and Oracle Pharma Marketing Analytics"](#)

### E.6.1 Oracle's Siebel Consumer Sector Dashboards and Pages

Oracle's Siebel Consumer Sector Sales Analytics extends the base Sales Analytics application to include Trade Funds Management Analytics, Trade Promotion Evaluation Analytics, Sales Performance Analytics, and Retail Audit Analytics.

All Consumer Sector specific metadata has been added to a single subject area. In the Oracle BI Administration Tool, this metadata is tagged with a red apple icon. The following topic covers each fundamental area of Consumer Sector Analytics and provides tips for data entry for effective analytics.

The Consumer Sector dashboards and pages available to the end user are described in [Table E-1](#).

**Table E-1 Consumer Sector Dashboards and Page Tabs**

Dashboard	Page	Function
Retail Audit	Last Audit	Shows the aggregated results of the last audit, defined the last time a product was audited at a specific account and merchandising location. The aggregated metrics can be drilled into to get a list of accounts or products to target future efforts.

**Table E-1 (Cont.) Consumer Sector Dashboards and Page Tabs**

Dashboard	Page	Function
	Trends	Displays key metrics captured in a retail audit over time across accounts or product categories.
	Position	Combines account and product performance with the individual representative responsible.
Promotion	Plan Year To Date	Displays both individual promotional performance with cumulative trends to provide overall perspective on meeting promotional targets.
	Key Accounts	Shows post promotion evaluation from a key account perspective across several levels of detail including plan, promotion, or promoted product detail.
	Corporate	Shows post promotion evaluation from a brand managers perspective, by evaluating corporate promotions.
Funds	Summary	Displays the key analyses for a fund manager, including Remaining Amounts which provides a status of all funds relative to all other funds.
	Accounts	Highlights status of funds and funds spent at a specific account and is targeted to assist key account managers in determining how to fund future promotions.
Sales Performance	Sales Volume Planning	Details the key metrics used in sales volume planning, including incremental, target, shipment and consumption volumes, over time and in comparison to one another. Baseline and estimated incremental volumes are stacked and compared to targets to assist in identifying future progress against targets. This analysis uses a global filter to assist end users in getting to the level of details they want in one selection.
	Hierarchy	Similar in content to Sales Volume Planning, however, this analysis is organized to promote exploration of data, allowing end users to freely drill up or down the account, time, or category product hierarchies. This page should be used to help sales managers identify where sales exceeded expectations.
	Trends	Depicts sales trends across accounts, channels, and categories as well as compares account performance in order to rank them.
	Growth	Displays key sales metrics versus the year ago and charts the rate of growth.
VP Sales	Business Overview	This page focuses on answering key business questions of a sales executive including where are my sales? How effective are my promotions by channel? Which account plans are top and bottom performers? How is my promoted volumes and funds spend trending as compared to last year? What are my top five accounts in each category?
	Product Overview	This page focuses on answering key product questions of a sales executive. For example, What percentage of total sales is in each category? What products are selling where by whom?
Key Account Manager	Account	This page focuses on answering key business questions of a key account manager. For example, How I am performing versus my account plan? What is my promotional forecast accuracy? What funds are available to plan additional retail activity?
	Category	This page focuses on answering key product questions of a key account manager. For example, Which category is best promoted at my accounts? How are my store conditions trending? Are out of stocks preventing me from reaching my targets?

## E.6.2 Oracle's Siebel Consumer Sector Data Requirements

The data requirements for the Consumer Sector–specific portion of the data model are detailed in [Table E-2](#). This includes the fund, promotion, performance and retail audit schema.

**Table E-2 Data Requirements for Consumer Sector Schema**

Page	Function
Funds	Trade Fund Management Analytics incorporates the latest Trade fund functionality including aggregation of approved adjustments, transfers, deal allocations, and payments at all statuses.
Promotion	In the transactional database, fund allocations and product movement (incremental volumes) can be assigned to a promotion at the promotion or the promoted product level. The ETL transfers this information at the promoted product level only. If you allocate funds at the promotion level and assign product detail such as shipment quantities at the promoted product level, the fund allocation data needs to be pushed to the promoted product level to be accurately stored in the Oracle Business Analytics Warehouse.
Performance	<p>Sales Performance Analytics is used primarily in conjunction with the category product hierarchy as defined by Oracle's Siebel Sales Volume Planning. To create the category-product hierarchy, the SVP adopt flag must be selected in order to aggregate product data up the category product hierarchy. This flag can be found by navigating to Catalog Administration, then SVP Category Details View, then Products list.</p> <p>The data warehouse and all prebuilt analyses are built from the end date of a period. To make sure data is accurate, the periods in the Oracle Business Analytics Warehouse tables must be of a single duration. If they are different, inaccurate data could be stored in the data warehouse. For example, if a week and a month end on the same date, the actual shipped quantities are combined during the building of the data warehouse.</p>
Retail Audit	<p>The Last Audit Flag is set during the ETL process. The flag is set to Yes for the most recent record for a product at an account and merchandising location. All other audit records for that combination of product, account, and merchandising location are set to No.</p> <p>The observation date is used to sort the retail audit records to determine which is the last audit. The observation date is populated when the status of a retail audit activity is changed to Done. The field observation date does not show up in the Oracle Business Analytics Warehouse user interface and does not have to be the same as the activity date.</p>

## E.6.3 Oracle's CME family of products (Oracle Communications, Media and Energy Sales Analytics)

Oracle's CME family of products (Oracle Communications, Media and Energy Sales Analytics,

Oracle Communications, Media and Energy Service Analytics, Oracle Communications, Media and Energy Marketing Analytics) makes use of order management functionality configured for CME. For Oracle's CME applications to fully reflect the information collected by CME order management functionality, some extensions to the Oracle CME Analytics application may be required. This topic explains these potential extensions.

Oracle's Siebel Sales Orders include complex products and simple products.

**Complex Products.** A series of products related by a product hierarchy. The highest product in the hierarchy is the root product, and the lower level products are the child products. In complex products, revenue figures are summed and roll up to the root product using the ROLLUP\_NET\_PRI field. For a complex product, Oracle Business

Intelligence examines only the root product when computing revenue. Child products are disregarded because their revenue is already reflected in the root.

**Simple Products.** A root product. Oracle Business Intelligence examines this root product when computing revenue, and nothing more.

Oracle's Siebel Communications, Media and Energy order management functionality supports products which have recurring charges over time (for example, \$20 per month for 12 months), one-time charges (for example, one-time purchase price of equipment), and usage charges (for example, 15 cents per minute).

The revenue attributed to a product with recurring charges is valued by taking the product's net price and multiplying it by the number of months that product is anticipated to be active, as represented by the Number of Revenue Occurrences field. This field, contained in Quote Item and Order Item records, is contained in the Oracle Business Analytics Warehouse by the following fields:

- W\_QUOTEITEM\_F.NUM\_OCCURRENCE
- W\_ORDERITEM\_F.NUM\_OCCURRENCE

In Oracle's CME family of products (Oracle Communications, Media and Energy Sales Analytics, Oracle Communications, Media and Energy Service Analytics, Oracle Communications, Media and Energy Marketing Analytics), revenue metrics do not automatically account for all recurring charges, and do not consider the NUM\_OCCURRENCE fields. Instead, Oracle's CME family of products revenue metrics incorporate one-time charges, one-month's worth of recurring charges, and no usage charges. To incorporate the anticipated value of all recurring charges, the W\_QUOTEITEM\_F.NUM\_OCCURRENCE and W\_ORDERITEM\_F.NUM\_OCCURRENCE fields may need to be incorporated into revenue calculations made during the Extraction, Transformation and Load (ETL) process for order item and line item records.

Alternatively, these fields in the Oracle Business Analytics Warehouse, representing the aggregated recurring and one-time product charges, may be used and incorporated into the ETL processes:

- S\_ORDERITEM.PER\_MTH\_CHG\_SUBTOT
- S\_ORDERITEM.ONETIME\_CHG\_SUBTOT
- S\_QUOTEITEM.PER\_MTH\_CHG\_SUBTOT
- S\_QUOTEITEM.ONETIME\_CHG\_SUBTOT

Each CME Order line item and Quote line item contains an Action Type of Add, Update, or Delete. Because Oracle Business Intelligence only looks at root product line items, only the Action Types associated with the root product are considered during analysis. Therefore, while all line items for a complex product may collectively include a combination of various Action Types, only the Action Type for the root product are considered during analysis. This is of special importance if a filter or query criteria in analysis is based on the Action Type field, which it is for most Account Management and Revenue Management dashboard reports.

Similarly, each CME Order line item and Quote line item is associated with a product of a particular Price Type. Because Oracle Business Intelligence considers root products only, only the Price Type associated with the root product is considered during analysis. Again, this is important if a filter or query criteria is based on Price Type. Such filter criteria apply to most Account Management and Revenue Management dashboard reports.

## E.6.4 Oracle's Siebel CME Dashboards and Pages

Oracle's Siebel Communications, Media and Energy (CME) Analytics contains corresponding industry-specific metadata. In the Oracle BI Administration Tool, industry-specific metadata is flagged with an icon picturing a telephone. Although this icon is visible in the Oracle BI Administration Tool, it is neither visible nor included within Oracle BI Answers. End users use Oracle BI Answers to access metadata for building queries and reports. For users of Oracle BI Answers to view and access CME metadata columns, they must log in using one of the CME responsibilities listed in [Table E-3](#). These responsibilities also determine what subject areas the user may access.

**Table E-3 Communications, Media and Energy Dashboards and Page Tabs**

Dashboard	Page	Function
Loyalty Management	Customer Lifetime Value	Segments customers based upon defined ranges of scores predicting customer lifetime value.
	Churn Propensity	Segments customers based on defined ranges of scores estimating churn propensity.
	Selling Propensity	Segments customers based on defined ranges of scores valuing the potential of up-sell and cross-sell opportunities.
	Financial Risk	Segments customers based on defined ranges of scores predicting financial risk.
	Actual Churn	Shows trending of actual customer churn, and actual customer acquisition, over time.
Revenue Management	Revenue Trends	Charts trends of order revenue and order volume over time, and identifies top products based on order revenue and volume.
	Service Activations	Charts trends of service activations over time, and indicates top service activations based on order revenue and volume.
	Service Modifications	Charts trends of service modifications over time, and indicates top service modifications based on order revenue and volume.
	Service Disconnections	Charts trends of service disconnections over time, and identifies services with the highest disconnection volume.
Account Management	Sales Portal	Identifies top accounts, and related top products, based upon order revenue and order volume.
	Service Activations	Charts trends of account service activations, and indicates top accounts based on service activation performance metrics.
	Service Modifications	Charts trends of account service modifications, and indicates top accounts based on service modification performance metrics.
	Service Disconnections	Charts trends of account service disconnections, and identifies accounts with the highest volume of disconnections.
	Trouble Tickets	Provides trouble ticket trending charts, and performance indicators, for particular accounts, and for accounts with selected common characteristics.
	Customer Satisfaction	Provides customer satisfaction trending charts, and performance indicators, for particular accounts, and for accounts with selected common characteristics.

## E.6.5 Oracle's Siebel CME Data Requirements

The data requirements for each of the Communications, Media and Energy dashboards are detailed in [Table E-4](#).

**Table E-4 Data Requirements for Communications, Media and Energy Dashboards**

Dashboard	Function
Loyalty Management	This dashboard uses customer scores generated from any third-party predictive modeling application offering the following predictive models: Customer Lifetime Value, Churn Propensity, Up-Sell Propensity, Cross-Sell Propensity, and Financial Risk Propensity. Scores must be generated for each Siebel Account, integrated into the Siebel Transaction Database, and then written to the Oracle Business Analytics Warehouse. This dashboard uses the Oracle Business Intelligence Customers subject area. See <a href="#">Section E.5, "Developing and Deploying Predictive Scores"</a> for more information.
Revenue Management	This dashboard uses data generated by Oracle's Siebel Communications, Media and Energy order management and interactive selling functionality. No specific data requirements are required beyond the data generated from these Siebel modules. This dashboard uses the Oracle Business Intelligence Orders and Products subject areas.
Account Management	This dashboard uses data generated by Oracle's Siebel Communications, Media and Energy order management, interactive selling, and service functionality. No specific data requirements are required beyond the data generated from these Siebel modules. This dashboard uses the Oracle Business Intelligence Orders, Products, Service Request and Customer Satisfaction subject areas.

## E.6.6 Oracle Pharma Sales Analytics Dimensions

Although the following dimensions are used in all subject areas, this topic describes the configuration necessary for Pharma Analytics applications. For more information, please refer to *Siebel Life Sciences Guide Version 8.0 Appendix B: Configuring Data for Siebel Pharma Analytics*.

### E.6.6.1 Positions Dimension

A sales territory is defined in Group Administration–Positions by a Siebel position. Creating parent positions creates the sales force hierarchy. Up to 10 levels of sales force hierarchy are supported by the application. Employees should be assigned to positions to populate employee hierarchy.

Position Types need to be set up according to compensation type (Rx or sales) only at the sales territory level. A district manager does not need to have a Position Type assigned to it. Sales Allocation needs to be exposed on the list to enter script compensation percentages (Rx or Sales) associated with each territory. For example, if all sales representatives receive 100% of the Rx on a ZIP Code, no action is needed or Position Type = Sales Representative can be assigned to the position.

Seed data on the Position Type list of values has been enhanced to include types for mirror, job share, and swat. Typically, both mirror and job share represent a position that receives less than 100% of the total scripts on a ZIP Code.

### E.6.6.2 Alignments Dimension

A sales territory alignment is the relationship of ZIP Code-to-territory or brick-to-territory. The alignment relationship is created in Oracle's Siebel Assignment Manager under Assignment Administration–Territories, as shown in [Table E-5](#).

**Table E-5 Sales Territory Alignment**

Relationship	Criteria	Comments
Contact ZIP to Territory	Contact ZIP Code	Use contact primary address ZIP Codes. Do not use ranges of ZIP Codes (that is, enter unique ZIP Codes as low and high values).  Do not enter duplicate ZIP Codes.
Account ZIP to Territory	Account ZIP Code	Do not use ranges of ZIP Codes (that is, enter unique ZIP Codes as low and high values).  Do not enter duplicate ZIP Codes.
Contact Brick to Territory	Contact Brick	Use contact primary address brick. Do not use ranges of bricks (that is, enter unique bricks as low and high values).  Do not enter duplicate ZIP Codes.
Account Brick to Territory	Account Brick	Do not use ranges of bricks (that is, enter unique bricks as low and high values).  Do not enter duplicate ZIP Codes.
Account to Territory	Account	Do not enter duplicate accounts.
Contact to Territory	Contact	Do not enter duplicate contacts.

### E.6.6.3 Products Dimension

The product hierarchy requires customer products (products of the company who licensed the software) to have predefined product types as shown in [Table E-6](#).

**Table E-6 Customer Products Predefined Product Types**

Product Level	Product Type	Example
3	Sample	Aracid 400 MG
2	Detail	Aracid
No Level	Sub Market	COPD
1	Market	Asthma

**Note:** Competitive products should use the product type Competitor. Competitor product hierarchies are set up using parent product relationships exclusively and should not have product levels assigned to them.

### E.6.6.4 Product Costs Dimension

Product costs for customer products (that is, products of the company that licensed the software) require population in the Product Administration, Product Form, as shown in [Table E-7](#).

**Table E-7 Product Costs For Customer Products**

Product Type	Field to be Populated
Sample	Sample Cost
Detail	Avg. Promo Cost

**Table E-7 (Cont.) Product Costs For Customer Products**

Product Type	Field to be Populated
Promotional Item Cost	Sample Cost

## E.6.7 Dimensions Specific to Subject Areas in Oracle Pharma Sales Analytics and Oracle Pharma Marketing Analytics

This section discusses the subject areas used by Pharma Analytics. For more information, please refer to *Siebel Life Sciences Guide Version 8.0 Appendix B: Configuring Data for Siebel Pharma Analytics*.

### E.6.7.1 Pharma Sales Effectiveness

This subject area is focused on syndicated data analytics.

The specific configuration required for the syndicated data depends on your data types, and the Analytics application and reports that you have licensed. The Data Loading Matrix table is the basis of prebuilt reports. The syndicated data loading matrix populates both base and derived metrics used in Pharmaceutical Sales Analytics.

### E.6.7.2 Pharma Product Categories

Oracle Pharma Sales Analytics and Oracle Pharma Marketing Analytics supports custom and prebuilt product category trees to allow roll-up of syndicated data by alternative hierarchies. To populate a custom category, first create a Catalog in Catalogue Administration, and create categories and subcategories as part of the catalogue. [Table E-8](#) lists the categories that need to have the Usage Type field populated in the Catalog Admin Category Detail list.

**Table E-8 Hierarchy Categories to be Populated in Pharma Analytics**

Usage Type Code	Hierarchy Category
ATC	Anatomical Therapeutic Class
Chemical	Chemical
Application Form	Product application
USC	User-defined codes and custom hierarchies

### E.6.7.3 Pharma Promotional Effectiveness

This subject area combines call activity data with syndicated data to analyze effectiveness of call activity.

Call Activity analysis records are derived from submitted call activity records stored in S\_EVT\_ACT in the Oracle Business Analytics Warehouse, where they are stamped with the ZIP Code or brick where the activity took place—that is, the Contact primary address's ZIP code/brick or the Account ZIP Code/brick. Allocation of these ZIP Code/brick records should be done by Assignment Manager rules to make sure that they are correctly allocated. Assignment Manager rules must match the Contact or Account primary address ZIP Codes or bricks. Otherwise, data integrity is not maintained.

Only calls that have status Submitted on the Pharma Professional Call Form are brought over from the Oracle Business Analytics Warehouse to the Oracle Business Analytics Warehouse.



**E.6.7.4 Pharma Medical Education Effectiveness**

This subject area combines measures from MedEd and Syndicated Data to measure effectiveness of medical education events used on Medical Education Analytics.

Only MedEd events with the status Completed on the Pharma ME Event List are extracted from Oracle Business Analytics Warehouse to populate the Oracle Business Analytics Warehouse.

MedEd Event costs are based on costs of activities in the Pharma ME Event Activity List. Costs are allocated based on MedEd Team cost allocation, and promoted products Cost Allocation on the MedEd event.

Costs are solely based on physician invitees with the status Attended in the Pharma ME Event Professional Invitee Session List.

Control groups are based on physicians who have the same contact ranking as attendee physicians within the same sales territory at the time of the event, but who did not attend the event.

**E.6.7.5 Pharma Objectives Achievement**

This subject is used to measure achievement and results for pharma call activity and Rx/sales targets. It is based on Pharma Objectives.

Objectives need to have a Unit populated in Retail Objective Form. Actual target numbers per contact and account need to be populated in the Pharma Campaign Target Account List or the Pharma Campaign Target Professional List Toggle.



---

---

# Oracle Business Analytics Warehouse for Life Sciences Data Considerations

This chapter contains the following topics:

- [Section F.1, "What's New for Pharma Analytics in Version 7.9"](#)
- [Section F.2, "Oracle Business Analytics Warehouse for Life Sciences Overview"](#)
- [Section F.3, "Importing Syndicated Data into Oracle Business Analytics Warehouse"](#)
- [Section F.4, "Syndicated Data Flat File Formats"](#)
- [Section F.5, "Life Sciences Data Loading Issues with Oracle Business Analytics Warehouse"](#)
- [Section F.6, "Incremental Updates in the Oracle Business Analytics Warehouse LS Dimension Tables"](#)

## F.1 What's New for Pharma Analytics in Version 7.9

This section describes the new features for Pharma Analytic, Version 7.9.

- New support for loading Syndicated Data with multiple Alignment Types.

Syndicated data can now be loaded with different Alignment Types to support the requirements of each Syndicated data type being loaded.

- Location of Alignment Version flag in the DAC has been changed.
  - The creation of historical Alignment Dimension hierarchy versions is controlled using a flag set within the DAC console. In Pharma Analytics 7.9 the location of this Alignment Version flag has moved to the Source System Parameters screen within the DAC's Design area.

## F.2 Oracle Business Analytics Warehouse for Life Sciences Overview

This section describes the key features and architecture of the Oracle Business Analytics Warehouse for Life Sciences Syndicated and Call Activity data.

- ETL for direct load of syndicated data.

Syndicated data can be directly loaded into the Oracle Business Analytics Warehouse.

A set of prebuilt processes against common external syndicated data types allows direct load of syndicated data, such as physician plan level Rx data, physician level Rx data, indirect sales data, and weekly Rx data.

- Syndicated data staging architecture.
  - Syndicated data is loaded against a staging architecture. The staging table architecture uses a denormalized time dimension that improves data loading by a factor of how many periods are loaded.
  - A set of cross-reference files for external key matching are supplied to resolve cross referencing and loading external syndicated data where the accounts, products, territories and contacts IDs are different from the Siebel IDs.
- Syndicated data and Call Activity data single grain fact tables.
  - Single grain syndicated data star schemas for power users and operational users.
  - Operational syndicated data star schemas for operational reporting.
  - Single grain call activity fact tables - account calls, account-attendee calls, and contact calls.
- Syndicated data Moving Annual Total (MAT) metrics calculated on ETL.
  - Calculation of rolling period calculations takes advantage of the syndicated data staging architecture to populate common pharmaceutical time aggregate metrics within the Oracle Business Analytics Warehouse.
- ETL for syndicated data market aggregation.
  - ETL process to load the Syndicated Data market fact tables based on aggregations on the product-market hierarchy. Minimizes the requirement to purchase pre-aggregated data by market.
- Set of prebuilt aggregation tables for high performance reporting.
  - Sales level aggregation tables for product-based syndicated data facts.
  - Sales level aggregation tables for market-based syndicated data facts.
  - Call activity aggregation tables for reports requiring top-level sales level activities.
  - De-duplication tables to de-duplicate data when the same contact is assigned to multiple territories for the district level for Physician Plan Level Rx syndicated data only.
- Mini dimensions.
  - Account and contact rank mini dimensions.
  - Representative specialty LOV dimensions.
  - W\_POSTN\_CON\_D used only for contact primary address.

### F.3 Importing Syndicated Data into Oracle Business Analytics Warehouse

Pharmaceutical companies purchase weekly and monthly sales and prescription data, known as syndicated data, from third-party vendors such as IMS, NDC, and Cegecim. Syndicated data vendors acquire data from drug wholesalers and retailers on a daily, weekly, and monthly basis and compile a master file of customers (wholesalers, pharmacies, hospitals, and doctors) and sales or prescription transactions for customers. Measures include indirect sales, indirect units, and prescriptions, and differ by vendor and periodicity. It is used for sales force analysis reporting and customer targeting.

The data is derived from panels of physicians, pharmacies, and so on, and projected nationally. Since the panels may change on a monthly basis, syndicated data suppliers tend to change their projections of sources on a monthly basis leading to full restatements of historical data. Thus, pharmaceutical companies are required to refresh fully the data in their data warehouses. In addition, weekly data requires incremental loading.

After it is delivered by the vendor, the syndicated data must be fully reloaded into the Oracle Business Analytics Warehouse in a timely manner and made available to users in order for them to make use of sales force analysis reporting.

This section includes the following topics:

- [Section F.3.1, "Syndicated Loading Definitions"](#)
- [Section F.3.2, "Data Types Supported in the Oracle Business Analytics Warehouse"](#)
- [Section F.3.3, "Loading Syndicated Data into the Oracle Business Analytics Warehouse"](#)

### F.3.1 Syndicated Loading Definitions

[Table F-1](#) provides terms and definitions related to syndicated loading.

**Table F-1 Syndicated Loading Definitions**

Term	Definition
Syndicated data	Third-party data from vendors that shows sales and prescription results for client defined markets.
Brick	Micro sales geographic area defined by vendor that contains prescription and sales trends for clients' product groupings or markets. Bricks do not exist in the US where the micro sales geographic area is commonly the postal code or zip code.
NRx	Abbreviation of new prescriptions. A new prescription is defined as dispensed prescriptions given a new number by the pharmacy, not necessarily new therapy for the patient.
TRx	Abbreviation of total prescriptions. $TRx = NRx + Refills$ . After the first time a prescription is filled, when it is refilled, the pharmacy refers back to the previous drug ID number and notes this as a refill.
Indirect sales	Total drug wholesalers product sales values to drug retailers (pharmacies) by brick or zip code. Sales values are calculated as units multiplied by client-selected price, where the price is the warehouse acquisition cost unless the client has chosen to have a different price applied to its direct sales or government depot sales.
Indirect units	Total drug wholesalers product sales units to drug retailers (pharmacies) by brick or zip code. Sales unit quantity is determined by multiplying package factor to obtain unit value (for example, one carton $\times$ 24 bottles = 24 units).

### F.3.2 Data Types Supported in the Oracle Business Analytics Warehouse

The Oracle Business Analytics Warehouse supports multiple data types defined by data source type and periodicity, as described in [Table F-2](#).

**Table F-2 Data Types Supported in Oracle Business Analytics Warehouse**

Data Type	Description
Prescription data by contact	Monthly NRx and TRx data for client defined markets that include competitor products data by physician.

**Table F–2 (Cont.) Data Types Supported in Oracle Business Analytics Warehouse**

<b>Data Type</b>	<b>Description</b>
Indirect sales brick level	Monthly wholesaler sales and unit values data for client defined markets that include competitor products aggregated by brick.
Indirect sales account level	Monthly wholesaler sales and unit values for company products by pharmacy.
Indirect sales zip level	Monthly wholesaler sales and unit values for client defined markets that include competitor products aggregated by postal code.
Direct sales account level	Monthly direct factory sales data and unit values to wholesalers for company's products by wholesaler.
Direct sales brick level	Monthly direct factory sales data and unit values to wholesalers for company's products by brick.
Direct sales zip level	Monthly direct factory sales data and unit values to wholesalers for company's products by postal code.
Weekly prescription data	Weekly NRx and TRx data for client defined markets that include competitor products by physician.
Plan level prescription data	Monthly prescription data by managed care plan for client defined markets that includes competitor products.
Sales market	<b>Incentives.</b> Monthly incentive compensation data sourced from internal incentive compensation systems and loaded as a new data type in the Oracle Business Analytics Warehouse.
Sales market	<b>Modified.</b> Monthly incentive compensation data sourced from internal incentive compensation systems and loaded as a new data type in the Oracle Business Analytics Warehouse.
Physician plan level Rx data	Monthly prescription data for physicians associated with a managed care plan.
Prescription data by zip code	Monthly prescription data for client defined markets that includes competitor products aggregated by postal code.
Prescription data by brick	Monthly prescription data for client defined markets that include competitor products aggregated by brick.

### F.3.3 Loading Syndicated Data into the Oracle Business Analytics Warehouse

The following options are available for loading syndicated data into the Oracle Business Analytics Warehouse:

- [Section F.3.3.1, "Loading From a Flat File Source Using Syndicated Data Staging Tables and Cross-Referencing Tables"](#)
- [Section F.3.3.2, "Loading From a Flat File Source Using Syndicated Market Staging Tables"](#)

#### F.3.3.1 Loading From a Flat File Source Using Syndicated Data Staging Tables and Cross-Referencing Tables

This option is supported in Oracle Business Intelligence Applications Version 7.8.3. To load syndicated data with this option, you have to prepare external data files for syndicated data and cross-reference data as described in tables [Table F–4](#) to [Table F–19](#) in section [Section F.4, "Syndicated Data Flat File Formats"](#).

The ETL process will load syndicated data source files as a full load each time, but cross-reference data files will be loaded incrementally. Cross-reference data files must contain only new and updated information each time to support proper incremental

loading. When the cross-reference data files are prepared, the data must be already loaded into the transactional database so the proper Siebel row IDs and the corresponding external source keys are resolved and provided in the data files.

Setting the correct alignment type to be used with the syndicated data is critical to loading data through the flat files. The Alignment Type is set in the external file AlignmentType.csv which is installed during the Oracle BI Applications installation in the folder. Before running the ETL, the Alignment Type must be set to one of the following options under ALIGN\_TYPE field.

- Zipcode
- Account/Contact
- Brick

Then, AlignmentType.csv file should be copied into the SrcFiles folder on Informatica Server machine, for example, C:\Program Files\Informatica PowerCenter 7.1.4\Server\SrcFiles.

In previous releases, only one Alignment Type could be set for use with all Syndicated Data Types. In Release 7.9, multiple Alignment Types can be set for different Syndicated Data Types.

For example, Physician Rx data can be loaded with an Account/Contact Alignment Type while Brick level Indirect Sales data can be loaded with a Brick Alignment Type. In addition, the same target Syndicated data table (for example the Physician Rx table) can be loaded with data defined in multiple Alignment Types such as both Account/Contact and Zipcode.

The Syndicated Data Type and Alignment Type rule mappings are set in the AlignmentType.csv file and this data is loaded into the W\_ALIGN\_TYPE\_G table at the start of the ETL process for use in the Syndicated data load. The format of the AlignmentType.csv file is shown below.

---



---

**Note:** Before running the ETL, the values in the 'ALIGN\_TYPE' column in this file should be updated to reflect the correct Alignment Types to be used with the appropriate Syndicated Data Types listed in the 'DATA\_SRC\_CD' column.

---



---

**Table F-3 AlignmentType.csv Mapping File**

ROW_WID	DATA_SRC_CD	ALIGN_TYPE
1	RXBrk	Brick
2	RXEVM	Zipcode
3	RXPT	Account/Contact
4	RXPrf	Account/Contact
5	RXSMI	Zipcode
6	RXSMM	Zipcode
7	RXXPT	Account/Contact
8	RXZip	Zipcode
9	SlsDirAct	Account/Contact
10	SlsIndAct	Account/Contact

**Table F-3 (Cont.) AlignmentType.csv Mapping File**

ROW_WID	DATA_SRC_CD	ALIGN_TYPE
11	SlsIndBrk	Brick
12	SlsIndZip	Zipcode

You can change the Alignment Types only when you choose to run the full ETL load process.

Once the syndicated data and cross-reference source files are prepared, then the data can be loaded directly into Oracle Business Analytics Warehouse staging tables. If any invalid data source type is used in the syndicated data files, then the ETL will load as it is. You have to prepare the correct data source type in the data files. Only the data that should not be nullable but is null in data files will be filtered out separately so you can review the rejected data for the next ETL run.

When the syndicated data is loaded from the Oracle Business Analytics Warehouse staging tables to target base tables, such as W\_SYNDD\_RX\_F, the external source keys are replaced with Oracle Business Intelligence keys and the data is loaded in the normalized format.

Then, based on what you defined as a primary alignment type above, the syndicated data in base target tables is replicated into the proper owner position and loaded into the main target tables at the territory level, such as W\_SYND\_RX\_T\_F, which contains all base table attributes plus ranking and position information.

**F.3.3.1.1 Flat File Data Population Rules for Populating TRx Data** This section provides information about various major columns in the W\_SYNDD\_RX\_F table related to populating TRx data.

- INTEGRATION\_ID
  - Data type is varchar(30).
  - Value can be any unique number or combination of values coming from the external file source with the appended postfix -##, such as -1, if the data is corresponding to the first month of bulk syndicated data, and -2 if the data is corresponding to the second month of bulk data, and so on.
  - INTEGRATION\_ID manipulation is done by LS\_EXP\_FLATFILE\_GENERATE\_INTEGRATION\_ID transformation. It is required because the incoming INTEGRATION\_ID in flat file represents 26 different bulk loads in denormalized format. Therefore, the format must be changed to normalized for the Oracle Business Analytics Warehouse target table.
- DATASOURCE\_NUM\_ID
  - ETL run specific (a numeric value that indicates the source for the data).
  - Used in standard Siebel mappings to ensure each record carries a value indicating its source.
  - Used in source qualifiers to ensure that lookups on dimension tables when populating fact tables area referencing data that was loaded from a uniform data source.
  - Data source number 1 is reserved for the Siebel transactional database. The external source should use a number other than 1.
  - For syndicated data flat file loads, the external source number (other than 1) is loaded into staging tables. When the data is loaded from the staging tables to



target tables and external source keys are matched to Siebel IDs, the data source number is converted as 1 in the target table.

- Combination of INTEGRATION\_ID and DATASOURCE\_NUM\_ID must be a unique value (mandated by a unique index on both columns).

This means that the system will not accept the same record from the same source with identical unique IDs.

- DATA\_LEVEL\_WID

- Stores a foreign key in the W\_LOV\_UD table for the record corresponding to that syndicated data source type for the data.
- Value comes from DATA\_SRC\_CD in the flat file. For example, in the table W\_SYNDD\_RX\_F, the proper data source type defined in flat file must be one of the following:
  - \* RXPrf
  - \* RXZip
  - \* RXBrk
  - \* RXSMI
  - \* RXSMM
- The standard mapping uses the incoming DATA\_SRC\_CD code to do a lookup against the W\_LOV\_UD table to secure the ROW\_WID of that data source type and inserts it into the W\_SYNDD\_RX\_F table DATA\_LEVEL\_WID column.

- PAYER\_TYPE\_WID

- Stores a foreign key in the W\_INS\_PLAN\_D table for the record corresponding to that payer type of the data such as Cash, Total.
- Value comes from PAYER\_TYPE\_ID in the flat file.

- PERIOD\_TYPE\_WID

- Stores a foreign key in the W\_LOV\_D table for the record corresponding to the period for the data.
- Value comes from PERIOD\_CD in the flat file, such as Month, Week.
- The standard mapping uses the incoming PERIOD\_CD code to do a lookup against the W\_LOV\_D table to secure the ROW\_WID of that period type and inserts it into the W\_SYNDD\_RX\_F table PERIOD\_TYPE\_WID column

- GEO\_WID

- GEO\_WID is a foreign key to the W\_GEO\_D dimension, which contains a geographical hierarchy based on zip code as the lowest level of detail. It is preconfigured to allow users to roll up a TRx measure to levels such as city, state or province, county, country, and continent.
- If the flat file source of TRx data load is zip code level, then the ZIPCODE field in the flat file must be provided. Then, the value is performed using the following logic to get loaded in the target GEO\_WID:
  - \* MPLT\_GEO\_WID\_ZIPCODE takes zip code as the incoming value, and the literal NO\_DUP\_FLG = 'N' ports to do a lookup against W\_GEO\_D for any record where the ZIPCODE column matches the incoming zip code value and the DUP\_ZIPCODE column = 'N'. (Sample data in the W\_GEO\_

D table contains multiple city records per zip code, only one of which is marked as DUP\_ZIPCODE = 'N'). The mapping also ensures that even with multiple matches only the first record is retrieved, and with no matches, the data comes back with the appropriate Unspecified code.

- CON\_GEO\_WID
  - CON\_GEO\_WID is a foreign key to the W\_GEO\_D dimension, which contains a geographical hierarchy based on zip code as the lowest level of detail.
  - The following three attributes, city, country, and zip code, are the main keys to determine CON\_GEO\_WID, and are brought from flat file CON\_CITY, CON\_COUNTRY, and CON\_ZIPCODE fields.
  - If the flat file source of TRx data load is contact level, then CON\_CITY, CON\_COUNTRY, and CON\_ZIPCODE fields in flat file must be provided. Then, the value is performed the following logic to get loaded in target CON\_GEO\_WID:
    - \* LS\_EXP\_FLATFILE\_CITY\_COUNTRY\_ZIP\_CHKNULL and LS\_MPLT\_FLATFILE\_GEO\_WID use the combination of city, country, and zip code to retrieve the first available record from W\_GEO\_D even if duplicates exist and regardless of the NO\_DUP\_FLG.

- ALIGN\_WID
  - Depending on your selected alignment item type, the following transformation is performed:

- \* When alignment item type is Zipcode:

```
IIF (DATA_SRC_CD = 'RXPrf', IN_CON_GEO_WID,
     IIF (DATA_SRC_CD = 'RXBrk' OR DATA_SRC_CD = 'RXSMI' OR DATA_SRC_CD =
          'RXSMM' OR DATA_SRC_CD = 'RXZip', IN_GEO_WID,
          ETL_UNSPEC_NUM))
```

- \* When alignment item type is Account/Contact:

```
IIF (DATA_SRC_CD = 'RXBrk' OR DATA_SRC_CD = 'RXPrf' OR DATA_SRC_CD =
     'RXSMI' OR DATA_SRC_CD = 'RXSMM' OR DATA_SRC_CD = 'RXZip', IN_CONTACT_
     WID, ETL_UNSPEC_NUM)
```

- \* When alignment item type is Brick:

```
IIF (DATA_SRC_CD = 'RXPrf', IN_CON_AREA_WID,
     IIF (DATA_SRC_CD = 'RXBrk' OR DATA_SRC_CD = 'RXSMI' OR DATA_SRC_CD =
          'RXSMM' OR DATA_SRC_CD = 'RXZip', IN_AREA_WID,
          ETL_UNSPEC_NUM))
```

The resulting value is used to populate the ALIGN\_WID column in the fact table, which should also be found in W\_ALIGNMT\_DH table ALIGN\_WID column.

Based on this ALIGN\_WID column, Oracle Business Analytics Warehouse replicates proper territory and populates W\_SYND\_RX\_T\_F table as well.

---



---

**Note:** This section discusses the W\_SYNDD\_RX\_F table as an example only. In the Oracle Business Analytics Warehouse there are six tables used for the different syndicated data sources which are loaded in the same manner: W\_SYNDD\_DS\_F, W\_SYNDD\_IDS\_F, W\_SYNDD\_RX\_F, W\_SYNDD\_PT\_F, W\_SYNDD\_XPT\_F, and W\_SYNDD\_W\_F tables.

---



---

### F.3.3.2 Loading From a Flat File Source Using Syndicated Market Staging Tables

The Oracle Business Analytics Warehouse supports loading syndicated market data using flat files. You have to prepare flat files of syndicated market data source and cross-referencing data source files as described in tables [Table F-4](#) to [Table F-19](#) in section [Section F.4, "Syndicated Data Flat File Formats"](#).

By default this option is disabled. To use this feature, you must turn on the option manually in the DAC Client. Turning on this option allows the flat file syndicated market data to be loaded directly into the Oracle Business Analytics Warehouse staging table W\_SYNDM\_RX\_FS, and then loaded into the target table W\_SYNDM\_RX\_F.

#### To load syndicated market data using flat files

1. Open the DAC Client, and navigate to Design, then Task.
2. Enter the following query:

```
LS Load into SyndicatedMarket Fact for*
```

Eleven tasks are returned.

3. Deactivate the following tasks, which, by default, are turned on:
  - LS Load into SyndicatedMarket Fact for Calculated Exponent Plantrak
  - LS Load into SyndicatedMarket Fact for Calculated Indirect Sales
  - LS Load into SyndicatedMarket Fact for Calculated Plantrak
  - LS Load into SyndicatedMarket Fact for Calculated Rx
  - LS Load into SyndicatedMarket Fact for Calculated Weekly Early View

---



---

**Note:** Do not deactivate LS Load into SyndicatedMarket Fact for Calculated Direct Sales.

---



---

4. Activate the following tasks:
  - LS Load into SyndicatedMarket Fact for Exponent Plantrak
  - LS Load into SyndicatedMarket Fact for Indirect Sales
  - LS Load into SyndicatedMarket Fact for Plantrak
  - LS Load into SyndicatedMarket Fact for Rx
  - LS Load into SyndicatedMarket Fact for Weekly Early View
5. Save your work.

## F.4 Syndicated Data Flat File Formats

This section includes descriptions of the following flat files used in the syndicated data load process:

- [Section F.4.1, "Syndicated Data Flat File for Direct Sales Data"](#)
- [Section F.4.2, "Syndicated Data Flat File for Indirect Sales Data"](#)
- [Section F.4.3, "Syndicated Data Flat File for Prescription Data"](#)
- [Section F.4.4, "Syndicated Data Flat File for Plan Level Rx Data"](#)
- [Section F.4.5, "Syndicated Data Flat File for Physician Plan Level Rx Data"](#)
- [Section F.4.6, "Syndicated Data Flat File for Weekly Early View Data"](#)
- [Section F.4.7, "Syndicated Market Flat File for Indirect Sales Data"](#)
- [Section F.4.8, "Syndicated Market Flat File for Prescription Data"](#)
- [Section F.4.9, "Syndicated Market Flat File for Plan Level Rx Data"](#)
- [Section F.4.10, "Syndicated Market Flat File for Physician Plan Level Rx Data"](#)
- [Section F.4.11, "Syndicated Market Flat File for Weekly Early View Data"](#)
- [Section F.4.12, "Flat File Cross-Reference Data Population Rules"](#)

### F.4.1 Syndicated Data Flat File for Direct Sales Data

[Table F-4](#) provides information about the syndicated data flat file for Direct Sales data.

**Table F-4** *Syndicated Data Flat File - Direct Sales Data*

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: SlsDirAct
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
AMT_DT	VARCHAR2(15)		Exchange date for currency code. Date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address for account level. <b>Note:</b> Corresponding Siebel S_REGION_EXT.ROW_ID (S_CON_ADDR.BRICK_ID) should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Initial capital format; for example, Short Hills.

**Table F-4 (Cont.) Syndicated Data Flat File - Direct Sales Data**

Field Name	Data Type	Field Required	Field Description
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code.
AMT_CURCY_CD	VARCHAR2(30)		Currency code for Amount.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on. <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
PROD_ID	VARCHAR2(30)		External product ID. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
S_AMT_M01	NUMBER(22,7)		Stores sales amounts of current month data.
S_AMT_M02	NUMBER(22,7)		Stores sales amounts of previous month data.
S_AMT_M03 - S_AMT_M06	NUMBER(22,7)		
S_UNIT_M01	NUMBER(22,7)		Stores sales units of current month data (for example, 20050301 data).
S_UNIT_M02	NUMBER(22,7)		Stores sales units of previous month data (for example, 20050201 data).
S_UNIT_M03	NUMBER(22,7)		Stores sales units of 20050101 data.
S_UNIT_M04 - S_UNIT_M026	NUMBER(22,7)		

## F.4.2 Syndicated Data Flat File for Indirect Sales Data

Table F-5 provides information about the syndicated data flat file for Indirect Sales data.

**Table F-5 Syndicated Data Flat File - Indirect Sales Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: SlsIndAct (Indirect sales account level) SlsIndZip (Indirect sales zip code level) SlsIndBrk (Indirect sales brick level)

**Table F-5 (Cont.) Syndicated Data Flat File - Indirect Sales Data**

Field Name	Data Type	Field Required	Field Description
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
AMT_DT	VARCHAR2(15)		Exchange date for currency code. Date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address for account level. <b>Note:</b> Corresponding Siebel S_REGION_EXT.ROW_ID (S_CON_ADDR.BRICK_ID) should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, Short Hills.
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
AMT_CURCY_CD	VARCHAR2(30)		Applies to syndicated data only. Currency code for Amount.
AREA_ID	VARCHAR2(30)		External syndicate brick ID. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on. <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
PROD_ID	VARCHAR2(30)		Applies to syndicated data only. External product ID. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
ZIPCODE	VARCHAR2(30)		Syndicate zip code. Enter value if DATA_SRC_CD is zip code level.

**Table F-5 (Cont.) Syndicated Data Flat File - Indirect Sales Data**

Field Name	Data Type	Field Required	Field Description
S_AMT_M01	NUMBER(22,7)		Stores sales amounts of current month data.
S_AMT_M02	NUMBER(22,7)		Stores sales amounts of previous month data.
S_AMT_M03	NUMBER(22,7)		Stores sales amounts of 20050101 data.
S_AMT_M04 - S_AMT_M026	NUMBER(22,7)		
S_UNIT_M01	NUMBER(22,7)		Stores sales units of current month data.
S_UNIT_M02	NUMBER(22,7)		Stores sales units of previous month data.
S_UNIT_M03	NUMBER(22,7)		Stores sales units of 20050101 data.
S_UNIT_M04 - S_UNIT_M26	NUMBER(22,7)		

### F.4.3 Syndicated Data Flat File for Prescription Data

Table F-6 provides information about the syndicated data flat file for Prescription data.

**Table F-6 Syndicated Data Flat File - Prescription Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	RXPrf (Prescription data by Contact) RXZip (Prescription data by Zip Code) RXBrk (Prescription data by Brick) RXSMI (Sales Market – Incentives) RXSMM (Sales Market – Modified)
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
AREA_ID	VARCHAR2(30)		External syndicate brick ID. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CONTACT_ID	VARCHAR2(30)		External contact ID. <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.
CON_AREA_ID	VARCHAR2(30)		External syndicate brick ID of contact's primary address. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.

**Table F-6 (Cont.) Syndicated Data Flat File - Prescription Data**

Field Name	Data Type	Field Required	Field Description
CON_CITY	VARCHAR2(50)		Contact city. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on.  <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code:  Month
PROD_ID	VARCHAR2(30)		External product ID.  <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
ZIPCODE	VARCHAR2(30)		Syndicate zip code. Enter value if DATA_SRC_CD is zip code level.
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
NRX_M01	NUMBER(22,7)		Stores New Rx of current month data.
NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data.
NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data.
NRX_M04 - NRX_M26	NUMBER(22,7)		
TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data.
TRX_M02	NUMBER(22,7)		Stores Total Rx of previous month data.
TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
TRX_M04 - TRX_M26	NUMBER(22,7)		

#### F.4.4 Syndicated Data Flat File for Plan Level Rx Data

Table F-7 provides information about the syndicated data flat file for Plan Level Rx data.



**Table F-7 Syndicated Data Flat File - Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXPT
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account and plan ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address. Enter value if DATA_SRC_CD is account level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, Short Hills
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
PROD_ID	VARCHAR2(30)		Applies to syndicated data only. External product ID. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
NRX_M01	NUMBER(22,7)		Stores New Rx of current month data.
NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data.
NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data.
NRX_M04 - NRX_M26	NUMBER(22,7)		
TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data.

**Table F-7 (Cont.) Syndicated Data Flat File - Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
TRX_M02	NUMBER(22,7)		Stores Total Rx of previous month data.
TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
TRX_M04 - TRX_M26	NUMBER(22,7)		

### F.4.5 Syndicated Data Flat File for Physician Plan Level Rx Data

Table F-8 provides information about the syndicated data flat file for Physician Plan Level Rx data.

**Table F-8 Syndicated Data Flat File - Physician Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXXPT
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Enter a value if DATA_SRC_CD is account level. Initial capital format; for example, Short Hills
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
CONTACT_ID	VARCHAR2(30)		External contact ID. <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.

**Table F–8 (Cont.) Syndicated Data Flat File - Physician Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
CON_AREA_ID	VARCHAR2(30)		External brick ID of contact's primary address. Enter value if DATA_SRC_CD is contact level.  <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CON_CITY	VARCHAR2(50)		Contact city. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
PROD_ID	VARCHAR2(30)		Applies to syndicated data only. External product ID. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
NRX_M01	NUMBER(22,7)		Stores New Rx of current month data.
NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data.
NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data.
NRX_M04 - NRX_M26	NUMBER(22,7)		
TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data.
TRX_M02	NUMBER(22,7)		Stores Total Rx s of previous month data.
TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
TRX_M04 - TRX_M26	NUMBER(22,7)		

#### F.4.6 Syndicated Data Flat File for Weekly Early View Data

[Table F–9](#) provides information about the syndicated data flat file for Weekly Early View data.

**Table F-9 Syndicated Data Flat File - Weekly Early View Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXEVM
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
CONTACT_ID	VARCHAR2(30)		External contact ID. <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.
CON_AREA_ID	VARCHAR2(30)		External brick ID of contact's primary address. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CON_CITY	VARCHAR2(50)		Contact city. Enter a value if DATA_SRC_CD is contact level. Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level. Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on. <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Week
PROD_ID	VARCHAR2(30)		Applies to syndicated data only. External product ID. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
NRX_W01	NUMBER(22,7)		Stores New Rx of current week data.

**Table F–9 (Cont.) Syndicated Data Flat File - Weekly Early View Data**

Field Name	Data Type	Field Required	Field Description
NRX_W02	NUMBER(22,7)		Stores New Rx of previous week data.
NRX_W03	NUMBER(22,7)		Stores New Rx of 20050101 data.
NRX_W04 - NRX_W13	NUMBER(22,7)		
TRX_W01	NUMBER(22,7)		Stores Total Rx of current week data.
TRX_W02	NUMBER(22,7)		Stores Total Rx of previous week data.
TRX_W03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
TRX_W04 - TRX_W13	NUMBER(22,7)		

## F.4.7 Syndicated Market Flat File for Indirect Sales Data

Table F–10 provides information about the syndicated market flat file for Indirect Sales data.

**Table F–10 Syndicated Market Flat File - Indirect Sales Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: SlsIndAct (Indirect sales account level) SlsIndZip (Indirect sales zip code level) SlsIndBrk (Indirect sales brick level)
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
MKT_AMT_DT	VARCHAR2(15)		Exchange date for currency code. Date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address for account level. <b>Note:</b> Corresponding Siebel S_REGION_EXT.ROW_ID (S_CON_ADDR.BRICK_ID) should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Enter value if DATA_SRC_CD is account level.  Initial capital format; for example, Short Hills.

**Table F–10 (Cont.) Syndicated Market Flat File - Indirect Sales Data**

Field Name	Data Type	Field Required	Field Description
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level.  Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
AREA_ID	VARCHAR2(30)		External syndicate brick ID. Enter value if DATA_SRC_CD is brick level.  <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
MARKET_ID	VARCHAR2(30)		External market ID for the product.  <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
MKT_AMT_CURCY_CD	VARCHAR2(30)		Currency code for Amount.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on.  <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
ZIPCODE	VARCHAR2(30)		Syndicate zip code. Enter value if DATA_SRC_CD is zip code level.
MKT_S_AMT_M01	NUMBER(22,7)		Stores sales amounts of current month data.
MKT_S_AMT_M02	NUMBER(22,7)		Stores sales amounts of previous month data.
MKT_S_AMT_M03	NUMBER(22,7)		Stores sales amounts of 20050101 data.
MKT_S_AMT_M04 - MKT_S_AMT_M26	NUMBER(22,7)		
MKT_S_UNIT_M01	NUMBER(22,7)		Stores sales units of current month data.
MKT_S_UNIT_M02	NUMBER(22,7)		Stores sales units of previous month data.
MKT_S_UNIT_M03	NUMBER(22,7)		Stores sales units of 20050101 data.
MKT_S_UNIT_M04 - MKT_S_UNIT_M26	NUMBER(22,7)		

### F.4.8 Syndicated Market Flat File for Prescription Data

Table F–11 provides information about the syndicated market flat file for Prescription data.

**Table F–11 Syndicated Market Flat File - Prescription Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	RXPrf (Prescription data by Contact) RXZip (Prescription data by Zip Code) RXBrk (Prescription data by Brick) RXSMI (Sales Market – Incentives) RXSMM (Sales Market – Modified)
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
AREA_ID	VARCHAR2(30)		External syndicate brick ID. Enter value if DATA_SRC_CD is brick level.  <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CONTACT_ID	VARCHAR2(30)		External contact ID.  <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.
CON_AREA_ID	VARCHAR2(30)		External syndicate brick ID of contact's primary address. Enter value if DATA_SRC_CD is brick level.  <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CON_CITY	VARCHAR2(50)		Contact city. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.
MARKET_ID	VARCHAR2(30)		External market ID for the product.  <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on.  <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.

**Table F-11 (Cont.) Syndicated Market Flat File - Prescription Data**

Field Name	Data Type	Field Required	Field Description
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
ZIPCODE	VARCHAR2(30)		Syndicate zip code. Enter value if DATA_SRC_CD is zip code level.
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
MKT_NRX_M01	NUMBER(22,7)		Stores New Rx of current month data.
MKT_NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data.
MKT_NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data.
MKT_NRX_M04 - MKT_NRX_M26	NUMBER(22,7)		
MKT_TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data.
MKT_TRX_M02	NUMBER(22,7)		Stores Total Rx of previous month data.
MKT_TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
MKT_TRX_M04 - MKT_TRX_M26	NUMBER(22,7)		

#### F.4.9 Syndicated Market Flat File for Plan Level Rx Data

Table F-12 provides information about the syndicated market flat file for Plan Level Rx data.

**Table F-12 Syndicated Market Flat File - Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_ NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXPT
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account and plan ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address. Enter value if DATA_SRC_CD is account level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.



**Table F-12 (Cont.) Syndicated Market Flat File - Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
ACCNT_CITY	VARCHAR2(50)		Account city. Enter value if DATA_SRC_CD is account level.  Initial capital format; for example, Short Hills
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level.  Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
MARKET_ID	VARCHAR2(30)		External market ID for the product.  <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code:  Month
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
MKT_NRX_M01	NUMBER(22,7)		Stores New Rx of current month data (for example, 20050301 data).
MKT_NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data (for example, 20050201 data).
MKT_NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data.
MKT_NRX_M04 - MKT_NRX_M26	NUMBER(22,7)		
MKT_TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data (for example, 20050301 data).
MKT_TRX_M02	NUMBER(22,7)		Stores Total Rx of previous month data (for example, 20050301 data).
MKT_TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
MKT_TRX_M04 - MKT_TRX_M26	NUMBER(22,7)		

#### F.4.10 Syndicated Market Flat File for Physician Plan Level Rx Data

Table F-13 provides information about the syndicated market flat file for Physician Plan Level Rx data.

**Table F-13 Syndicated Market Flat File - Physician Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.

**Table F-13 (Cont.) Syndicated Market Flat File - Physician Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXXPT
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
ACCNT_ID	VARCHAR2(30)		External account ID. <b>Note:</b> Corresponding Siebel S_ORG_EXT.ROW_ID should be provided in cross-reference file.
ACCNT_AREA_ID	VARCHAR2(30)		External brick ID of account's primary address. Enter value if DATA_SRC_CD is brick level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
ACCNT_CITY	VARCHAR2(50)		Account city. Enter a value if DATA_SRC_CD is account level. Initial capital format; for example, Short Hills
ACCNT_COUNTRY	VARCHAR2(30)		Account country. Enter value if DATA_SRC_CD is account level. Initial capital format; for example, France; or abbreviate format, such as USA.
ACCNT_ZIPCODE	VARCHAR2(30)		Account zip code. Enter value if DATA_SRC_CD is account level.
CONTACT_ID	VARCHAR2(30)		External contact ID. <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.
CON_AREA_ID	VARCHAR2(30)		External brick ID of contact's primary address. Enter value if DATA_SRC_CD is contact level. <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CON_CITY	VARCHAR2(50)		Contact city. Enter value if DATA_SRC_CD is contact level. Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level. Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.

**Table F–13 (Cont.) Syndicated Market Flat File - Physician Plan Level Rx Data**

Field Name	Data Type	Field Required	Field Description
MARKET_ID	VARCHAR2(30)		External market ID for product. <b>Note:</b> Corresponding Siebel S_PROD_INT.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code: Month
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
MKT_NRX_M01	NUMBER(22,7)		Stores New Rx of current month data (for example, 20050301 data).
MKT_NRX_M02	NUMBER(22,7)		Stores New Rx of previous month data (for example, 20050201 data).
MKT_NRX_M03	NUMBER(22,7)		Stores New Rx of 20050101 data (for example, 20050301 data).
MKT_NRX_M04 - MKT_NRX_M26	NUMBER(22,7)		
MKT_TRX_M01	NUMBER(22,7)		Stores Total Rx of current month data.
MKT_TRX_M02	NUMBER(22,7)		Stores Total Rx of previous month data.
MKT_TRX_M03	NUMBER(22,7)		Stores Total Rx of 20050101 data.
MKT_TRX_M04 - MKT_TRX_M26	NUMBER(22,7)		

#### F.4.11 Syndicated Market Flat File for Weekly Early View Data

Table F–14 provides information about the syndicated market flat file for Weekly Early View data.

**Table F–14 Syndicated Market Flat File - Weekly Early View Data**

Field Name	Data Type	Field Required	Field Description
DATASOURCE_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
DATA_SRC_CD	VARCHAR2(30)	NOT NULL	Enter the following Siebel data source type code: RXEVM
INTEGRATION_ID	VARCHAR2(25)	NOT NULL	External integration ID.
START_DT	VARCHAR2(15)	NOT NULL	Syndicate data date format YYYYMMDD.
CONTACT_ID	VARCHAR2(30)		External contact ID. <b>Note:</b> Corresponding Siebel S_CONTACT.ROW_ID should be provided in cross-reference file.

**Table F-14 (Cont.) Syndicated Market Flat File - Weekly Early View Data**

Field Name	Data Type	Field Required	Field Description
CON_AREA_ID	VARCHAR2(30)		External brick ID of contact's primary address. Enter value if DATA_SRC_CD is brick level.  <b>Note:</b> Corresponding Siebel S_REGION.ROW_ID should be provided in cross-reference file.
CON_CITY	VARCHAR2(50)		Contact city. Enter a value if DATA_SRC_CD is contact level.  Initial capital format; for example, Short Hills
CON_COUNTRY	VARCHAR2(30)		Contact country. Enter value if DATA_SRC_CD is contact level.  Initial capital format; for example, France; or abbreviate format, such as USA.
CON_ZIPCODE	VARCHAR2(30)		Contact zip code. Enter value if DATA_SRC_CD is contact level.
PAYER_TYPE_ID	VARCHAR2(30)		External payer type ID. This column is used to define payment type, such as Cash, Government, Third Party, Total, and so on.  <b>Note:</b> Corresponding Siebel S_INS_PLAN.ROW_ID should be provided in cross-reference file.
PERIOD_CD	VARCHAR2(30)		Enter the following Siebel period code:  Week
CONV_FACTOR	NUMBER(22,7)		Applies to Rx data only. Used to convert Rx volume to measurable units which will be used to dollarize the Rx volume.
MKT_NRX_W01	NUMBER(22,7)		Stores New Rx of current data.
MKT_NRX_W02	NUMBER(22,7)		Stores New Rx of previous data.
MKT_NRX_W03	NUMBER(22,7)		Stores New Rx of 20050101 data.
MKT_NRX_W04 - MKT_NRX_W14	NUMBER(22,7)		
MKT_TRX_W01	NUMBER(22,7)		Stores Total Rx of current week data.
MKT_TRX_W02	NUMBER(22,7)		Stores Total Rx of previous week data.
MKT_TRX_W03	NUMBER(22,7)		Stores Total Rx of 20050214 data.
MKT_TRX_W04 - MKT_TRX_W13	NUMBER(22,7)		

#### F.4.12 Flat File Cross-Reference Data Population Rules

The following tables provide information about flat file cross-reference data population rules:

- [Table F-15, " Flat File Cross-Reference Data Population Rules for Area \(Brick\) Data"](#)

- [Table F–16, " Flat File Cross-Reference Data Population Rules for Account and Plan Data"](#)
- [Table F–17, " Flat File Cross-Reference Data Population Rules for Payer Type Data"](#)
- [Table F–18, " Flat File Cross-Reference Data Population Rules for Contact Data"](#)
- [Table F–19, " Flat File Cross-Reference Data Population Rules for Product and Market Data"](#)

**Table F–15 Flat File Cross-Reference Data Population Rules for Area (Brick) Data**

Field Name	Data Type	Field Required	Field Description
EXT_DATASRC_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.
EXT_INTEGRATION_ID	VARCHAR2(30)	NOT NULL	External area (brick) ID. <b>Note:</b> This same ID should be used in ACCNT_AREA_ID, AREA_ID, AND CON_AREA_ID syndicated data and syndicated market flat files.
EXT_AREA_NAME	VARCHAR2(50)		Area name. <b>Note:</b> The same value should already be loaded in Siebel S_REGION.NAME.
EXT_PROVINCE	VARCHAR2(50)		Area province. <b>Note:</b> The same value should already be loaded in Siebel S_REGION.PROVINCE.
EXT_STATE	VARCHAR2(50)		Area state. <b>Note:</b> The same value should already be loaded in Siebel S_REGION.STATE. Use the format all capitals and abbreviated, such as NJ.
EXT_COUNTRY	VARCHAR2(30)		Area country. <b>Note:</b> The same value should already be loaded in Siebel S_REGION.COUNTRY. Use the initial capital format; for example, France.
INTEGRATION_ID	VARCHAR2(30)	NOT NULL	Siebel area (brick) ID. <b>Note:</b> This value should come from Siebel S_REGION.ROW_ID.

**Table F–16 Flat File Cross-Reference Data Population Rules for Account and Plan Data**

Field Name	Data Type	Field Required	Field Description
EXT_DATASRC_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1. <b>Note:</b> The same value should be used in the DATASOURCE_NUM_ID field of the syndicated data and syndicated market flat files.

**Table F-16 (Cont.) Flat File Cross-Reference Data Population Rules for Account and**

Field Name	Data Type	Field Required	Field Description
EXT_INTEGRATION_ID	VARCHAR2(30)	NOT NULL	External account and plan ID. <b>Note:</b> The same ID should be used in ACCNT_AREA_ID, AREA_ID, AND CON_AREA_ID syndicated data and syndicated market flat files.
EXT_ACCNT_NAME	VARCHAR2(100)		Account and plan name. <b>Note:</b> The same value should already be loaded in Siebel S_ORG_EXT.NAME.
EXT_ACCNT_NUM	VARCHAR2(30)		Account outlet number.
INTEGRATION_ID	VARCHAR2(30)	NOT NULL	Siebel account ID. <b>Note:</b> This value should come from Siebel S_REGION.ROW_ID.

**Table F-17 Flat File Cross-Reference Data Population Rules for Payer Type Data**

Field Name	Data Type	Field Required	Field Description
EXT_DATASRC_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1. <b>Note:</b> The same value should be used in the DATASOURCE_NUM_ID field of the syndicated data and syndicated market flat files.
EXT_INTEGRATION_ID	VARCHAR2(30)	NOT NULL	External payer ID. <b>Note:</b> The same value should be used in the PAYER_TYPE_ID field of the syndicated data and syndicated market flat files.
EXT_PLAN_TYPE	VARCHAR2(30)		Plan type, such as Payer, Plan, and so on. <b>Note:</b> The same value should already be loaded in Siebel S_INS_PLAN.PLAN_TYPE.
EXT_PAYER_NAME	VARCHAR2(100)		Payer type, such as Cash, Government, Third Party, Total, and so on. <b>Note:</b> The same value should already be loaded in Siebel S_INS_PLAN.NAME.
INTEGRATION_ID	VARCHAR2(30)	NOT NULL	Siebel payer ID. <b>Note:</b> This value should come from Siebel S_REGION.ROW_ID.

**Table F–18 Flat File Cross-Reference Data Population Rules for Contact Data**

Field Name	Data Type	Field Required	Field Description
EXT_DATASRC_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.  <b>Note:</b> The same value should be used in the DATASOURCE_NUM_ID field of the syndicated data and syndicated market flat files.
EXT_INTEGRATION_ID	VARCHAR2(30)	NOT NULL	External contact ID.  <b>Note:</b> The same value should be used in the CONTACT_ID field of the syndicated data and syndicated market flat files.
EXT_CON_FULL_NAME	VARCHAR2(102)		Contact name.  <b>Note:</b> The same value should already be loaded in Siebel S_CONTACT.LAST_NAME and S_CONTACT.FST_NAME.
EXT_ME_NUM	VARCHAR2(50)		Contact ME number.  <b>Note:</b> The same value should already be loaded in Siebel S_CONTACT.CSN.
INTEGRATION_ID	VARCHAR2(30)	NOT NULL	Siebel contact ID.  <b>Note:</b> This value should come from Siebel S_CONTACT.ROW_ID.

**Table F–19 Flat File Cross-Reference Data Population Rules for Product and Market Data**

Field Name	Data Type	Field Required	Field Description
EXT_DATASRC_NUM_ID	NUMBER(10)	NOT NULL	External data source number ID other than 1.  <b>Note:</b> The same value should be used in the DATASOURCE_NUM_ID field of the syndicated data and syndicated market flat files.
EXT_INTEGRATION_ID	VARCHAR2(30)	NOT NULL	External product and market ID.  <b>Note:</b> The same value should be used in the PAYER_TYPE_ID field of the syndicated data and syndicated market flat files.
EXT_PROD_TYPE	VARCHAR2(30)		Product and market type.  <b>Note:</b> The same value should already be loaded in Siebel S_PROD_INT.
EXT_PROD_NAME	VARCHAR2(50)		Product and market name.  <b>Note:</b> The same value should already be loaded in Siebel PROD_INT.NAME.
INTEGRATION_ID	VARCHAR2(30)	NOT NULL	Siebel product ID.  <b>Note:</b> This value should come from Siebel S_PROD_INT.ROW_ID.

## F.5 Life Sciences Data Loading Issues with Oracle Business Analytics Warehouse

This issue is specific to Analytics for Life Sciences and does not affect other products.

The ETL process updates the Oracle Business Analytics Warehouse for Life Sciences with a full refresh or using incremental updates. In DAC, you can run a full load by selecting Tools, then ETL Management, then Reset Data Warehouse. This procedure is usually only used for the initial build. If you run the same execution plan without confirming the Reset Data Warehouse again, then the ETL incrementally updates the Oracle Business Analytics Warehouse.

Running a full refresh load by confirming the Reset Data Warehouse feature deletes all existing information stored in the fact and dimension tables. The following types of incremental update rules are supported on the fact tables.

- **Account Call Fact.** Adds new records (call activity for account) to the fact table.
- **Attendee Call Fact.** Adds new records (call activity for attendee) to the fact table.
- **Contact Call Fact.** Adds new records (call activity for contact) to the fact table
- **Syndicated Data - Base Direct Sales Fact, Territory Direct Sales Fact, Base Indirect Sales Fact, Territory Indirect Sales Fact, District Indirect Sales Fact, Base Physician Rx Fact, Territory Physician Rx Fact, Base Rx Plan Fact, Territory Rx Plan Fact, Base Weekly Physician Rx Fact, Territory Weekly Physician Rx Fact, Base Physician Rx Plan Fact, Territory Physician Rx Plan Fact.** ETL does not support incremental updates. When running the full refresh ETL, all records in the fact and dimension tables are deleted. To maintain a history in the dimension tables (such as multiple alignments), use the incremental ETL. If you need to incrementally update the syndicates data fact tables for incremental syndicated data loading, use one of the following strategies:
  - **For incremental insert.** Prepare flat file source data that has new data with the new INTEGRATION\_ID. Load the data directly into the staging table and modify the session not to truncate the fact tables. Then use the existing ETL to load data into fact tables.
  - **For incremental update.** Create new mappings that do a lookup against the fact tables and run the update. Make sure that INTEGRATION\_ID used in the flat file source and in the target tables are not identical. Because the incoming syndicated data in the flat file going to the target tables is in the normalized format, the INTEGRATION\_ID must be manipulated properly. However, the constant rule is applied when the INTEGRATION\_ID is manipulated during the ETL load. The syndicated data during the first bulk load has the 'original in-coming INTEGRATION\_ID' || '-1'; the 26th bulk load has the 'original in-coming INTEGRATION\_ID' || '-26'.

### F.5.1 Known Issues with the Syndicated Data Fact Tables

The following are known issues with creation of aggregate measure columns in the Syndicated Data fact tables.

- With large volumes of syndicated data, the creation of aggregate measures in the Oracle Business Analytics Warehouse can take four times the amount of time needed to load the fact table. This may be unacceptable under circumstances when there are large amounts of syndicated data.
  - Incremental Updates in the Oracle Business Analytics Warehouse LS Dimension Tables.



- MAT aggregate measures are handled by the ETL process and not by metadata. All other aggregate measures are handled by metadata.

## F.6 Incremental Updates in the Oracle Business Analytics Warehouse LS Dimension Tables

This issue is specific to Oracle Business Analytics Warehouse for Life Sciences and does not affect other products. In the LS dimension tables the following incremental updates are supported. Some dimensions have more than one set of attributes.

### F.6.1 W\_ALIGNMT\_DH

**Base Dimensional Hierarchy Table.**

**Attribute.** Zip code, Brick, or Account/Contact Hierarchy depends on what alignment item type is selected.

---



---

**Note:** In release 7.9, all Zip Code, Brick and Account/Contact hierarchy alignment can be loaded.

---



---

**Description.** Contains the current version of alignment only.

### F.6.2 W\_ALIGNVER\_DH

**Slowly Changing Dimension Type.**

**Attribute.** Historical Zip code, Brick, or Account/Contact Hierarchy depends on what alignment item type is selected.

---



---

**Note:** In release 7.9, all Zip Code, Brick and Account/Contact hierarchy alignment can be loaded.

---



---

**Description.** Tracks historical alignments. Note the following:

- Depending on selected alignment type in AlignmentType.csv, only the selected alignment rules are extracted and loaded into the Oracle Business Analytics Warehouse. You can change the alignment type only when you run a full refresh ETL.

---



---

**Note:** In release 7.9, all Zip Code, Brick and Account/Contact alignment types can be extracted and loaded.

---



---

- When the alignment rule is loaded into the Oracle Business Analytics Warehouse, the current alignment rule is stored in the W\_ALIGNMT\_DH base table and the existing historical alignment rules are stored in W\_ALIGNVER\_DH table.
- The creation of Alignment Versions is controlled through the setting of the Alignment Version flag in the DAC console. In Pharma Analytics 7.9 this flag is now set in the Source System Parameters screen within the DAC's Design area. The parameter name is '\$\$ALIGNMENT\_VER' and takes a value of either Y or N.

- If the \$\$ALIGNMENT\_VER parameter is set to N, then the ETL makes changes to existing alignments if there is a change on an assignment rule.
- If the \$\$ALIGNMENT\_VER parameter is set to Y, it creates a new alignment version. Any existing alignment moves to the W\_ALIGNVER\_DH table as history. The W\_ALIGNVER\_DH table is used only when the \$\$ALIGNMENT\_VER parameter is set to Y.

The dimension has these characteristics:

- The first alignment after Full load is 1.
- Incremental Updates in the Oracle Business Analytics Warehouse LS Dimension Tables
- A new version is created when a new assignment rule (new position or position relationship) or an assignment criteria is modified (change in postal code, brick, contact, or account) if the Alignment Version flag is set to Y.
- Assignment criteria:
  - Contact ZIP Code or Account ZIP Code cannot use ZIP Code ranges. Each ZIP Code assigned to a territory needs to be on a separate row, so the same value needs to be entered for both ZIP Code Low and ZIP Code High. For example, if ZIP Code 09654 is assigned to a territory, the value for both ZIP Code High and ZIP Code Low should be 09654. Also, the same ZIP Code should not be assigned twice to the same territory and a ZIP Code should be unique to a territory when using assignment criteria for Contact ZIP Code or Account ZIP Code.
  - Contact Brick or Account Brick require unique bricks assigned to a territory.
  - Every new version increases the counter by one and is a sequential number. Any pre-existing history version alignment data moves to W\_ALIGNVER\_DH historical alignment table whenever a new version is created. That is, the W\_ALIGNMT\_DH table always maintains the latest version while the W\_ALIGNVER\_DH table maintains the rest of the historical version if the \$\$ALIGNMENT\_VER parameter is set to Y. So, when the historical alignment dimension table gets too large, your administrator can create a SQL query to delete all attributes for version numbers from the W\_ALIGNVER\_DH table.
  - Effective date of the alignment is assignment activation date.

---

---

# Pharma Disconnected Analytics Administration - Supplemental Information

This appendix describes additional configuration steps for Oracle's Pharma Disconnected Analytics, and contains the following sections:

- [Section G.1, "About Pharma Disconnected Analytics"](#)
- [Section G.2, "Sourcing Reports for Oracle's Pharma Disconnected Analytics"](#)

## G.1 About Pharma Disconnected Analytics

Pharma Disconnected Analytics is a preconfigured Disconnected Analytics application for Pharma Sales. Pharma components for Disconnected Analytics are installed during an Oracle Business Intelligence installation in the \OracleBIData\Disconnected\Pharma directory.

The following is a list of components that have been tailored for Pharma Disconnected Analytics and their locations:

- **PharmaDisconnect.rpd.** The Pharma Disconnected Analytics repository is tailored for Pharma Disconnected Analytics and downloaded to the laptop during synchronization. It is located in the \OracleBIData\Disconnected\Pharma\Application directory.
- **PharmaDisconnect.webcat.** Presentation Services tailored for Pharma Disconnected Analytics. It is located in the \OracleBIData\Disconnected\Pharma\Application directory.
- **Pharma.XML.** Application configuration file tailored for Pharma Disconnected Analytics. It is located in the \OracleBIData\Disconnected\Pharma directory.
- **SQL Files.** Series of predefined DDL files to create the Pharma Disconnected Analytics mobile database tables and indexes. The files are located in the \OracleBIData\Disconnected\Pharma\Application directory.
- **Messages XML Files.** Language specific message XML files for localized Presentation Catalog captions in the Pharma Disconnected Analytics application. The files are located in the \OracleBIData\Disconnected\Pharma\messages directory.
- **Disconnected Sourcing Reports.** Predefined Presentation Catalog reports used to extract data during the synchronization process to populate the tables in the Pharma Disconnected Application's mobile database. The reports are located in the Presentation Services catalog under the shared\disconnected folder.

The Pharma Disconnected Analytics components have been built to work against the standard server side Oracle Pharma Sales Analytics Application and Data Warehouse. The Pharma Disconnected Analytics components need to be customized to correctly reflect any changes and customizations made to the standard server side application before any initial synchronization.

To improve the performance of preprocessed synchronization by reducing query execution time on the Oracle BI Server, you can fine-tune the index strategy and sort the data during the ETL mapping process.

---

**Note:** For more detailed information on Disconnected Analytics Application configuration and deployment, see the *Oracle Business Intelligence Disconnected Analytics Administration and Configuration Guide*.

---

## G.2 Sourcing Reports for Oracle's Pharma Disconnected Analytics

Table G–1 contains a list of the sourcing reports (queries) for Oracle's Pharma Disconnected Analytics. These reports issue logical SQL to the Oracle BI Server to extract the data used to populate the local Disconnected client schema.

**Table G–1 Sourcing Reports for Pharma Disconnected Analytics**

Subject Area	Sourcing Report Name	Target Table
Pharma Promotional Effectiveness	Day Dimension	W_DAY_D
Pharma Promotional Effectiveness	Product Ranking Dimension	W_CON_RANK_D
Pharma Promotional Effectiveness	Brick Dimension	W_AREA_D
Pharma Promotional Effectiveness	Geography Dimension	W_GEO_D
Pharma Promotional Effectiveness	Payor Plan Dimension	W_INS_PLAN_D
Pharma Promotional Effectiveness	Product Hierarchy	W_PRODUCT_DH
Pharma Promotional Effectiveness	Position Hierarchy	W_POSITION_DH
Pharma Promotional Effectiveness	Call Priority List of Value Dimension	W_LOV_D
Pharma Promotional Effectiveness	Product Indication List of Value Dimension	W_LOV_D
Pharma Sales Effectiveness	Period Type List of Value Dimension	W_LOV_D
Pharma Promotional Effectiveness	Person Dimension	W_PERSON_D
Pharma Objective Achievement	Plan Promotion Dimension	W_PLAN_PROMO_D
Pharma Customer Demographics	Contact Primary Address Dimension	W_POSTN_CON_D
Pharma Promotional Effectiveness	Contact Call Activity Fact	W_CON_CALL_F

**Table G-1 (Cont.) Sourcing Reports for Pharma Disconnected Analytics**

Subject Area	Sourcing Report Name	Target Table
Pharma Promotional Effectiveness	Contact Call Activity Aggregate	W_CON_CALL_N_A
Pharma Objective Achievement	Contact Objective Fact	W_CON_OBJ_F
Pharma Sales Effectiveness	Indirect Sales Market Level Fact	W_MARKET_IDS_F
Pharma Sales Effectiveness	Indirect Sales Market Level Aggregate	W_MARKET_IDS_N_A
Pharma Sales Effectiveness	Prescription Market Level Fact	W_MARKET_RX_F
Pharma Sales Effectiveness	Prescription Market Level Aggregate	W_MARKET_RX_N_A
Pharma Sales Effectiveness	Direct Sales Fact	W_SYND_DS_F
Pharma Sales Effectiveness	Direct Sales Aggregate	W_SYND_DS_N_A
Pharma Sales Effectiveness	Indirect Sales Fact	W_SYND_IDS_F
Pharma Sales Effectiveness	Indirect Sales Aggregate	W_SYND_IDS_N_A
Pharma Sales Effectiveness	Prescription Sales Fact	W_SYND_RX_F
Pharma Sales Effectiveness	Prescription Sales Aggregate	W_SYND_RX_N_A

**Note:** The ATC hierarchy's ATC IV level should be linked to the lowest level of the Standard Product hierarchy and the lowest product level available for the Syndicated Data type loaded into the fact table. As a general rule, load data at the lowest level possible in the product hierarchy. For more information about best practices when using reports in Oracle's Pharma Analytics, see Appendix B in *Siebel Life Sciences Guide*, Version 8.0.



---

---

# Using Oracle Business Analytics Warehouse Exception Reports

---

---

**Note:** Exception Reports are only supported with Siebel CRM sources.

---

---

This chapter covers exception reports, which provide information about the source data used for ETL processes that can lead to erroneous results in the Oracle Business Analytics Warehouse or may cause data loss during the ETL process. These reports point out some of the known problematic areas, but they should not be relied upon to find all potential data inconsistencies in the source data.

This chapter includes the following topics:

- [Section H.1, "Understanding Oracle Business Analytics Warehouse Exceptions"](#)
- [Section H.2, "Executing Oracle Business Analytics Warehouse Exception Reports"](#)

## H.1 Understanding Oracle Business Analytics Warehouse Exceptions

Exception reports are defined for the following components:

- **List of Values.** Identifies gaps and overlaps for certain LOV types.
- **Cost Lists.** Identifies products for which the cost lists have not been defined. Identifies the cost lists which define costs for a product in a certain currency for overlapping periods of time.
- **Exchange Rates.** Currency Exchange rates that do not change over a period of time. If exchange rates are not defined for more than 30-day intervals, then they are flagged as an exception.
- **Hierarchies.** Entities that have circular references are flagged as exceptions. The Oracle Business Analytics Warehouse supports 10 levels of hierarchies. If there are entities that have more than 10 levels of hierarchies defined, they are flagged as exceptions.

### H.1.1 List of Values Exceptions

List of Values include High and Low values that can be used as bucket values in categories for effective analysis. If these values are not contiguous (such as gaps or overlaps in defined values), the ETL process cannot accurately categorize the values.

An example of List of Values exceptions is shown in [Table H-1](#).

**Table H-1 List of Values Exceptions (Example)**

Type	Name	Low	High
MY_TYPE	< 100	10	100
MY_TYPE	80 – 200	80	200
MY_TYPE	250 – 300	250	300
MY_TYPE	350 – 400	350	400

Notice that, in the example for Type MY\_TYPE, there are overlaps and gaps between records. There is an overlap of ranges between the first and second row. There is a gap between second and third row, and between third and fourth rows.

The following LOV types are analyzed for List of Value Exceptions:

- ACCNT\_REVENUE\_SIZE
- ACCNT\_EMP\_SIZE
- LEAD\_AGE\_DAYS
- OPTY\_REVENUE\_SIZE
- OPTY\_UNIT\_SIZE
- ACCNT\_REVENUE
- QUOTE\_AGE\_DAYS
- ACCNT\_REVN\_GROWTH
- APPROVAL\_AUTH\_SIZE
- SR\_CHART\_AGE
- ASSET\_COST\_CATEGORY

## H.1.2 Cost List Exceptions

Cost Lists for specified products and currency should not have overlapping time periods. If multiple cost lists are defined for a product and currency during a given time period, then the cost for the product may not be computed correctly in the Oracle Business Analytics Warehouse.

An example of Cost List exceptions is shown in [Table H-2](#).

**Table H-2 Cost List Exceptions (Example)**

Cost List	Product Name	Currency	Start Date (MM-DD-YYYY)	End Date (MM-DD-YYYY)	Cost
Cost List 1	Product 1	USD	01-01-2000	12-31-2000	10.00
Cost List 2	Product 1	USD	06-01-2000	06-01-2001	12.00
Cost List 3	Product 1	USD	06-01-2001	06-01-2002	13.00

In the example, Cost List 1 and 2 have definitions of cost overlapping over 06-01-2000 to 12-31-2000.



### H.1.3 Products Without a Cost List

During the ETL process, the costs of the products are calculated based on the Cost List table. If the cost lists are not defined correctly, the cost of the products cannot be calculated correctly in the Oracle Business Analytics Warehouse. This exception mapping queries the product table and looks for a minimum of one cost list to be defined. The products with no cost list definition are flagged as exceptions.

### H.1.4 Exchange Rate Exceptions

The Oracle Business Analytics Warehouse supports transactions in many different currencies. Oracle Business Intelligence converts all currencies in the Oracle Business Analytics Warehouse to a single currency for analysis purposes. The ETL Base Exchange Currency parameter in System Preferences, indicates the currency to which all the financial amounts will be converted. The Exchange rates are derived from the Exchange Rate tables in the Oracle Business Analytics Warehouse. If the currency exchange rates do not change for a period of 30 days, then Oracle Business Intelligence flags it as an exception.

If there are time period gaps in the exchange rate data, the ETL process defaults to the most recent recorded exchange rate. If the actual exchange rate is significantly more or less favorable than what is recorded in the database, the outdated exchange rate distorts the true value of currency amounts in the Oracle Business Analytics Warehouse.

---



---

**Note:** Exchange rates are derived from records that are of type 'Daily' in the Oracle Business Analytics Warehouse. If any other types have been defined, they are not handled without some customization.

---



---

### H.1.5 Invalid Hierarchy Exceptions

Accounts, divisions, products, and opportunities can all have hierarchical relationships. These entities are denormalized within the Oracle Business Analytics Warehouse database to a fixed number of levels. Oracle Business Intelligence supports up to ten hierarchies levels in the Oracle Business Analytics Warehouse. If the depth of hierarchies extends beyond this number, results become inconsistent or incomplete.

### H.1.6 Circular Hierarchy Exceptions

Circular Hierarchies arise when the parent-child relationship has circular references. See [Table H-3](#) for an example.

**Table H-3** *Circular Hierarchy Example 1*

Child	Parent
A1	A2
A2	A1

Oracle Business Intelligence flags exceptions for two levels. Circular references over two hierarchies are not flagged. See [Table H-4](#) for an example.

**Table H-4** *Circular Hierarchy Example 2*

Child	Parent
A1	A2

**Table H-4 (Cont.) Circular Hierarchy Example 2**

Child	Parent
A2	A3
A3	A1

These produce infinite levels of hierarchies. The same records will be captured under the Invalid Hierarchy exceptions as their hierarchy depths will increase beyond 10 levels.

## H.2 Executing Oracle Business Analytics Warehouse Exception Reports

Before loading the Oracle Business Analytics Warehouse for the first time and for the subsequent refreshes, you should plan to spend time cleansing your transactional database data using the exception reports. The process is iterative, and requires coordination with other team members who have responsibility for data in the transactional database, such as the Siebel database administrator. After the initial cleansing, you should generate the exception reports on a scheduled basis to maintain the integrity of your data.

---



---

**Note:** Rerunning the exception reports overwrites all data contained in this view.

---



---

To cleanse the Oracle Business Analytics Warehouse, repeat these actions until the exception report is empty:

- In DAC, run the Exception Reports execution plan.
- In Oracle's Siebel application that you are using, navigate to Analytics Administration, then Exception Reports.

---



---

**Note:** In Siebel Financial Services, this screen is called DataMart Administration.

---



---

- For every line in the ETL Exception Reports list, fix the cause of the problem. For information on fixing problems, see [Section H.2.1, "Cleansing Data"](#).

### H.2.1 Cleansing Data

Use a combination of the Exception Reports and the Diagnostic views to assess changes that need to be made external to the Oracle Business Analytics Warehouse, and changes to the Oracle Business Analytics Warehouse directly.

The ETL Exception Reports list contains one record for each exception flagged in the Oracle Business Analytics Warehouse. The ETL Exception Explanation form, located below the ETL Exception Reports list, describes the selected exception, its effect on the Oracle Business Analytics Warehouse building process, and offers suggestions for repairing the data.

#### To fix an exception

1. Select an exception record.
2. Read and understand the text in the ETL Exception Explanation form.

3. Click the report link.  
The object's data appears. (For example, if the object is an account, then the Account form appears. If the object is a cost list, then the Cost List list appears.)
4. Repair the problem, using the text in the ETL Exception Explanation form as a guide.
5. Return to the ETL Exception Reports list and place a check mark in the Fixed column to indicate to others that this exception has now been fixed.

## H.2.2 Using the List of Values View

Use the List of Values view, shown in the figure below, to visually compare how the list of values data extracted from the Oracle Business Analytics Warehouse coordinates with the values loaded into the Oracle Business Analytics Warehouse. The ETL process removes duplicates and overlaps and fills data gaps. Values are extended to span the List of Values (LOV) minimum and maximum values. Duplicates, Range Gaps, and Overlaps are flagged by the exception reports.

Figure H-1 List of Values View

Type	Language Independent Code	Display Value	Low	High	Order	Language Name
ABS_COST_FCT_AREA	Field Service	Field Service			1	English-American
ABS_COST_FCT_AREA	Professional Services	Professional Services			2	English-American
ABS_COST_FCT_AREA	Workforce Management	Workforce Management			3	English-American
ABS_COST_FCT_TYPE	Normal	Normal			1	English-American
ABS_COST_FCT_TYPE	Emergency	Emergency			2	English-American
ABS_COST_FCT_VARIABLE	Constraint Violation	Constraint Violation			1	English-American
ABS_COST_FCT_VARIABLE	FSE Overtime	FSE Overtime			2	English-American

Type	Language Independent Code	Display Value	Low	High	Order
ABS_COST_FCT_AREA	Field Service	Field Service	0	9,999,999,999	1
ABS_COST_FCT_AREA	Professional Services	Professional Services	0	9,999,999,999	2
ABS_COST_FCT_AREA	Workforce Management	Workforce Management	0	9,999,999,999	3
ABS_COST_FCT_TYPE	Normal	Normal	0	9,999,999,999	1
ABS_COST_FCT_TYPE	Emergency	Emergency	0	9,999,999,999	2
ABS_COST_FCT_VARIABLE	Constraint Violation	Constraint Violation	0	9,999,999,999	1
ABS_COST_FCT_VARIABLE	FSE Overtime	FSE Overtime	0	9,999,999,999	2

The top List of Values list shows values from the Oracle Business Analytics Warehouse and the bottom List of Values (Data Warehouse) list shows the data that is to be used in ETL process. You can edit the Oracle Business Analytics Warehouse data directly in this view, but the Oracle Business Analytics Warehouse list is read-only.

**Note:** The List of Values is extracted into the Oracle Business Analytics Warehouse where the language is the same as the ETL Default Language set in the DAC Source System Parameters, or whose translate flag is set to 'N,' or those that are active. For more information about setting DAC Source System Parameters, see .

## H.2.3 Using the Exchange Rates View

Use the Exchange Rates view to diagnose currency translation issues in the Oracle Business Analytics Warehouse. The ETL process removes duplicates, fills gaps, and

removes overlaps. The ETL process computes exchange rates based on commutative and associative properties, such as product and reverse rates.

The top Exchange Rates list shows currencies, the middle Exchange Rates list shows the Oracle Business Analytics Warehouse values for active currencies and their exchange rates, and the bottom Exchange Rates (Data Warehouse) list shows the values loaded into the Oracle Business Analytics Warehouse for the selected currency in the upper Exchange Rates list to the ETL Base Exchange Currency. The Exchange Rates (Data Warehouse) list is read-only.

- The Active Currencies predefined query restricts the list to the active currencies in the Oracle Business Analytics Warehouse.
- The exception reports flag any exchange rates to the ETL Base Exchange Currency that have not been defined within a specified period (30 days) in the DAC Source System Parameters.

## H.2.4 Using the Cost List View

Use the Cost List view to display the cost lists from the Oracle Business Analytics Warehouse from the point of view of the product, and a read-only view of the values to be loaded into the Oracle Business Analytics Warehouse. The ETL process removes duplicates, overlaps, and fills gaps.

The Cost List list (top) shows products, and the Cost List Line Items list (middle) shows the cost lists associated with the selected product. The Cost Lists (Data Warehouse) list (bottom) shows the data as it is transformed for the Oracle Business Analytics Warehouse.

- The exception reports flag products that do not appear in the Cost List list or have Cost List time gaps and overlaps.
- The Oracle Business Analytics Warehouse contains only one Cost List for a product and a currency at a time.

## H.2.5 Using the ETL History View

After all of the exceptions are corrected, the building of the data warehouse can be initiated. This view lists the history of the ETL processes and their statuses. When each ETL batch starts, the name of the process along with the timestamp is set, the status is set to STARTED. When the batch completes, its status is updated to COMPLETED.

## H.2.6 Additional Exceptions

Additional exceptions include:

- The Analysis start and end date in the DAC Source System Parameters must span the entire period of time during which the transactions have occurred. For example, you may want to choose an early and late date range to cover the entire time period you are analyzing. These dates in the DAC Source System Parameters are crucial for the building of Day Dimension, flattening of Exchange Rates, Cost Lists, and KPI (Key Performance Indicator fact) calculations.
- The DAC Source System Parameters — ETL Date Format, ETL Analysis Start, ETL Analysis End parameters, and the List of Values — ETL\_UNSPEC\_DATE must be defined in the same data format. If one is changed, the others must be changed accordingly.

- List of Values must be defined appropriately. If there is no appropriate entry in List of Values, the strings that depend on List of Values in the Oracle Business Analytics Warehouse will not be translated.
- There must be exchange rates defined for the currencies your organization deals with. If the appropriate exchange values are not found, the ETL process uses the ETL Unknown Exchange Rate defined in the DAC Source System Parameters.



---

---

# Troubleshooting Oracle BI Applications

This chapter details some common problems related to the Oracle Business Analytics Warehouse and suggests possible solutions. It contains the following topics:

- [Section I.1, "Oracle Business Analytics Warehouse Installation and Initial Configuration Issues"](#)
- [Section I.2, "Informatica and Loading Issues"](#)
- [Section I.3, "Unicode Issues with Informatica and the Oracle Business Analytics Warehouse"](#)
- [Section I.4, "UNIX Installation Issues with the Oracle Business Analytics Warehouse"](#)
- [Section I.5, "Oracle Business Analytics Warehouse Upgrade Issues"](#)
- [Section I.6, "Oracle BI Interactive Dashboard Issues"](#)
- [Section I.7, "ETL Issues"](#)

**Note:** When you troubleshoot installation and configuration issues, use the following log information:

On Windows:

- the Informatica Server Log (check the Informatica Event Viewer)
- the Informatica Rep Server Log (check the Informatica Event Viewer)
- the Informatica Repagent Log (check the Informatica Event Viewer)
- the workflow log (located in ..\Informatica\Server\WorkFlowLogs)
- the session log for the session that fails (located in ..\Informatica\Server\SessLogs)
- the (Teradata) loader logs (located in ..\Informatica\Server\Tgtfiles)
- the Data Warehouse Application Console logs (located in the \DAC\log directory)

On UNIX:

- the Informatica Server Log (located in ..informatica/server/server.log)
- the Informatica Rep Server Log (located in ..informatica/repservers/repservers.log)
- the Informatica Repagent Log (located in ..informatica/repservers/reagent.log)
- the workflow log (located in ../informatica/Server/WorkFlowLogs)
- the session log for the session that fails (located in ../informatica/Server/SessLogs)

- the (Teradata) loader logs (located in ../informatica/Server/Tgtfiles)
- the Data Warehouse Application Console logs (located in the /DAC/log directory)

## I.1 Oracle Business Analytics Warehouse Installation and Initial Configuration Issues

[Table I-1](#) provides information about problems and solutions related to the installation and initial configuration of the Oracle Business Analytics Warehouse.

**Table I-1 Oracle Business Analytics Warehouse Installation and Configuration Issues**

Symptom/Error Message	Probable Cause/Solution
Cannot connect to the Oracle Business Analytics Warehouse running on DB2.	Make sure that the DB2 configuration matches the requirements listed in <a href="#">Section 3.3, "IBM DB2 UDB-Specific Database Guidelines for Oracle Business Analytics Warehouse"</a> .
After installing Informatica's ODBC driver (V3.5) on a Windows computer, you cannot open the Query Analyzer, and the Enterprise Manager shows an error message.	The ODBC driver installation replaces the odbcbcp.dll file with an older version. The Informatica documentation contains the procedure for correcting this problem.
Cannot connect to Oracle Business Analytics Warehouse from Informatica.	Make sure that the server running the Informatica Server software has an ODBC connection to the Oracle Business Analytics Warehouse using a Siebel ODBC driver and an ODBC connection to the Oracle Business Analytics Warehouse using the Merant Closed 3.5 32-bit driver.
Error 2140 (Informatica service failed to start).	The server may not be started or the Informatica Service may not be started. See Informatica's installation and configuration guide on Siebel eBusiness Third-Party Bookshelf for detailed information.
Informatica installation fails with an Unhandled Exception error and displays a message similar to this: 'Error Number: 0x80040707. Description: DLL function call crashed: ISRT._DoInstall.'	The computer is probably running out of virtual memory. Restart the computer and reinstall Informatica.
After selecting 'Warehouse Default Amount (Yen)' as the currency symbol, the dollar sign (\$) is still displayed.	The currencies.xml file must be edited. For instructions, see <a href="#">Appendix C.3, "Changing the Default Currency in Analytics Applications"</a> .

## I.2 Informatica and Loading Issues

[Table I-2](#) provides information about problems and solutions related to issues with Informatica and loading. To view the Informatica log file details, double-click the workflow.

**Table I-2 Informatica and Loading Issues**

Symptom/Error Message	Probable Cause/Solution
Double-clicking the workflow yields an error message in Informatica PowerCenter Workflow Manager: 'The system cannot find the file specified.'	The session log files are not set up properly. You also may need to change the text editor.



**Table I-2 (Cont.) Informatica and Loading Issues**

Symptom/Error Message	Probable Cause/Solution
Using Oracle, some mappings hang while running when performance statistics are switched on.	When running some Informatica mappings for loading the Oracle Business Analytics Warehouse, turning on the Performance Statistics can cause the mapping to hang. The only workaround is to increase the values of the LMSHaredMemory and MaxSessions variables in Informatica. The risk of increasing the LMSHaredMemory too much is that it may start to have a serious effect on overall performance of the machine that the Informatica server is running on.
When you execute a workflow in Informatica PowerCenter Workflow Manager, Informatica returns the following error message:  'Request to start workflow (workflow name) on server (server name) not successful.'	This can happen due to a server time-out property that is usually set to 20 or 40 seconds. When you try to run a large workflow, every session in that workflow is fetched into the server's memory. If this takes longer than the server time-out property, the server returns a message that the workflow was unable to run. However, the workflow is running, but the server just needs time to complete fetching the sessions into memory. Double-click the workflow to view the log file details.
Informatica (RDBMS is DB2) gives the following error message:  Error occurred unlocking [SDE_ServiceRequestDimension1].  An error occurred while accessing the repository[[IBM][CLI Driver][DB2/6000] SQL0955C  Sort memory cannot be allocated to process the statement. Reason code = ". SQLSTATE=57011]  DB2 Fatal Error[FnName: ExecuteDirect -- SQLSTATE=57011 [IBM][CLI Driver][DB2/6000]	The DB2 parameter "SHEAPTHRES" is too small.
When loading the data warehouse, Informatica reports a lock problem.	Either someone has a session open or there is a dead session. Make sure no one has any open sessions. If no sessions are open, then follow the Informatica documentation on removing locks caused by dead sessions.
Session SDEINC_RecordExtractStart fails due to unique constraint error while executing a Refresh workflow.	This could be because the previous load or refresh did not complete successfully. Fix the problem that caused the previous refresh session to fail. Make sure you start the process from last entry of %RestartNextWorkflow # before the failed session, and restart the workflow process from that point.
The session fails and you receive the following error code:  Error "TE_7042 Aggregate Error: File Operation Error	This is due to a disk space limitation. Check the /Informatica/PowerMart/Cache/Check directory for available disk space, also check the limits (ulimit) of the account used to start PowerMart.

**Table I-2 (Cont.) Informatica and Loading Issues**

Symptom/Error Message	Probable Cause/Solution
Informatica sessions get deadlocked and eventually fail when they try to do a 'select' from the repository table OPB_OBJECT_LOCKS. This problem sometimes occurs on MSSQL server databases.	<p>This is possibly caused by a limited number of resources on the MSSQL Database Server. The workaround is to execute the following MSSQL specific SQL command on the Oracle Business Analytics Warehouse:</p> <pre>DROP INDEX OPB_OBJECT_LOCKS.OPB_OBJ_LOCKS_IDX DROP INDEX OPB_OBJECT_LOCKS.OPB_OBJ_LOCKS_IDX2 DROP INDEX OPB_OBJECT_LOCKS.OPB_OBJ_LOCKS_IDX3</pre> <p>Upon completion of executing these commands, continue executing the workflow processes to load the Oracle Business Analytics Warehouse.</p>
An error may occur when trying to send a post session email notification using MS Outlook 2000. Refer to Informatica release notes for further information.	<p>After installing Informatica Server on Windows, copy the file mapi32.dll from winnt\system32 to the bin folder where the Informatica Server is installed, overwriting the existing mapi32.dll in that directory. Start the Informatica Server so that the Informatica Server can use the new mapi32.dll.</p> <p>The Extended MAPI Error. MAPILogonEx failed[2147746065] error indicates that the logon is not configured correctly. Check the following:</p> <ol style="list-style-type: none"> <li>1. Choose Services, then Informatica, then Logon, and make sure the login (domain\username) and password are correct.</li> <li>2. Choose Control Panel, then Mail (it may also be called Mail and Fax or Exchange), then Services, then Show Profiles, and make sure the mail profile is correct.</li> <li>3. Choose Programs, then Informatica Server, then Informatica Server Setup, then Miscellaneous, and make sure the MS Exchange profile is correct.</li> </ol>
While creating a custom session, bulk load mode does not work properly with SQL Server.	Change the mode to 'normal' in Informatica Repository for the session. The 'normal' mode must be used everywhere for SQL Server in all of your custom sessions.
Outlook closes when sending out a notification of finishing the ETL process.	Informatica is closing Outlook. This issue is known to Informatica and is scheduled to be resolved in an upcoming release. Until then, create a second profile in Outlook and add that profile name to the Informatica server setup.

**Table I-2 (Cont.) Informatica and Loading Issues**

Symptom/Error Message	Probable Cause/Solution
<p>Oracle 8i sessions running in bulk mode fail and return an error message similar to:</p> <pre>WRITER_1_1_1&gt; CMN_1022 Database driver error... CMN_1022 [ ORA-00600: internal error code, arguments: [kpodpmop_01], [2], [], [], [], [], [], [] Or MAPPING&gt; TE_7022 TShmWriter: Initialized MAPPING&gt; Sat Jan 26 13:54:45 2002 MAPPING&gt; TE_7001 Internal error: Failed to allocate a target slot. Sat MAPPING&gt; Jan 26 13:54:45 2002 TE_ 7017 Failed to Initialize Server MAPPING&gt; Transformation BLK_ALL_DATATYPES1 Sat Jan 26 13:54:45 2002MAPPING&gt; TM_6006 Error initializing DTM for session...MAPPING&gt; TM_6020 Session [s_BULK_LONG] completed at [Sat Jan 26 13:54:45 2002]</pre> <p>During an ETL execution, when Informatica and DAC Servers use DB2 Connect version 7 to talk to DB2/390 version 7 OLTP and data warehouse databases, you receive an error message similar to the following:</p> <pre>SEVERE: [IBM][CLI Driver][DB2] SQL0191N Error occurred because of a fragmented MBCS character. SQLSTATE=22504 103 SEVERE Tue May 11 21:37:29 CDT 2004 [IBM][CLI Driver][DB2] SQL0191N Error occurred because of a fragmented MBCS character. SQLSTATE=22504 [IBM][CLI Driver] CLI0102E Invalid conversion. SQLSTATE=07006 sqlstate = 07006 [IBM][CLI Driver][DB2/AIX64] SQL0303N A value cannot be assigned to a host variable in the SELECT, VALUES, or FETCH statement because the data types are not compatible. SQLSTATE=42806 sqlstate = 42806This issue comes up when you use 'Unicode' data movement in Informatica.</pre>	<p>This problem is an Oracle 8i defect. It has been resolved in Oracle 9i. The workaround is to run the session in Normal mode. To do so, in Informatica PowerCenter Workflow Manager, navigate to the Targets dialog, and change the Target Load type to Normal.</p> <p>This problem is a DB2 Connect version 7 (IBM) defect related to code page conversion. The problem has been resolved in DB2 Connect version 8.</p> <p>To correct the problem, do the following:</p> <ol style="list-style-type: none"> <li>Download the file IBM01140.ucs from <a href="ftp://ftp.software.ibm.com/ps/products/db2/fixes/english/siebel/siebel7/Conversion_Files">ftp://ftp.software.ibm.com/ps/products/db2/fixes/english/siebel/siebel7/Conversion_Files</a> to the /sqllib/conv directory.</li> <li>Make a copy of the file and rename it to IMB05348.ucs.</li> </ol> <p>The target table that the mapping is inserting into is a non-unicode table. Make sure the table is created as an Unicode table.</p>

**Table I-2 (Cont.) Informatica and Loading Issues**

Symptom/Error Message	Probable Cause/Solution
<p>The ELT run fails with one of the following errors:</p> <pre>FATAL ERROR : Caught a fatal signal [11]. Will abort this server process shortly. INFO : CMN_1131 : (3644 1) System shared memory [360454] for [Load Manager Shared Memory] removed.  FATAL ERROR : Aborting this server process due to a fatal signal [11].</pre>	<p>If you want to perform ETL runs in Informatica on HP-UX Itanium 2 64-bit, you need to install one of the following patches:</p> <p>HP-UX 11.11 Itanium 64-bit systems require Patch PHSS_33033 or higher.</p> <p>HP -UX 11.23 Itanium 64-bit systems require Patch PHSS_33349 or higher.</p>

### I.3 Unicode Issues with Informatica and the Oracle Business Analytics Warehouse

Table I-3 provides information about problems and solutions related to Unicode issues with Informatica and the Oracle Business Analytics Warehouse.

**Table I-3 Unicode Issues with Informatica and Oracle Business Analytics Warehouse**

Symptom/Error Message	Probable Cause/Solution
<p>Task fails and generates the error message: 'TRANSF_1_1_1_1&gt; TE_7073 Aggregate Error: Expecting keys to be ascending.' Tasks known to fail are SDE_DtlForecastFact and SDE_CostList</p>	<p>In Informatica PowerCenter Designer, open the corresponding mapping and in the Aggregator transformation, remove the check from the Sortedinput check box.</p>
<p>In the Windows 2000 environment, with a double-byte language, such as Japanese, when you try to input the path for the directory where Informatica is installed for the value \$PMRootDir in the Informatica PowerCenter Workflow Manager, the double-byte characters are converted to question marks. This causes the ETL to fail because it cannot find the Informatica Server folder and its subdirectories.</p>	<p>Share the folder under a new name with no double-byte characters, and use the network share as the value for \$PMRootDir.</p>

**Table I-3 (Cont.) Unicode Issues with Informatica and Oracle Business Analytics**

Symptom/Error Message	Probable Cause/Solution
The database connection code page is incorrect for Unicode/Non-Latin code pages.	<p>The database connection code page has to be set manually in Informatica PowerCenter Workflow Manager.</p> <ol style="list-style-type: none"> <li>1. In Informatica PowerCenter Workflow Manager, connect to the Oracle_BI_DW_Base using Administrator as the user.</li> <li>2. Click Menu Connection, then Select Relational.</li> <li>3. Select the appropriate OLTP connection (DB2_OLTP for DB2, ORA_OLTP for Oracle, MSSQL_OLTP or ODBC_OLTP for MSSQL).</li> <li>4. For the code page select the UTF-8 encoding of Unicode or the appropriate code page, and then click OK.</li> <li>5. Select the OLAP connection (DB2_OLAP for DB2, ORA_OLAP for Oracle, MSSQL_OLAP or ODBC_OLTP for MSSQL).</li> <li>6. For the code page select the UTF-8 encoding of Unicode or the appropriate code page, and then click OK.</li> </ol>
Unable to install Informatica PowerMart on non-English UNIX locales (for example, ESN, DEU, FRA, ITA). The following error message appears: 'FATAL ERROR during initialization, Invalid language specified, Application terminated.'	<p>This error will occur when the locale environment variables are set inconsistently or not at all. These settings are determined by entering the command <code>locale</code> at the UNIX command prompt. Make sure the correct locales are installed on the UNIX machine. Set the environment variable <code>PM_CODEPAGENAME</code> correctly (For example set it to <code>MS932</code> for Japanese).</p>
Symptom/Error Message: Unable to install Informatica PowerMart on non-English UNIX locales (for example, ESN, DEU, FRA, ITA). The following error message appears: 'FATAL ERROR during initialization, Invalid language specified, Application terminated.'	<p>Set the <code>LC_ALL</code> setting. For example:</p> <pre>LANG=en_US export LANG LC_ALL=en_US export LC_ALL</pre>

## I.4 UNIX Installation Issues with the Oracle Business Analytics Warehouse

Table I-4 provides information about issues related to installing the Oracle Business Analytics Warehouse on UNIX.

**Table I-4 UNIX Installation Issues with Oracle Business Analytics Warehouse**

Symptom/Error Message	Probable Cause/Solution
<p>When restoring the Informatica Repository, the following error message appears in the dwdb.log:</p> <pre>ERROR: OBJM_54543 [Tue Mar 25 17:47:16 2003]: (164   2484) DataBase error: ORA-01461: can bind a LONG value only for insert into a LONG column</pre>	<p>The database is Unicode and the environment variable is needed.</p> <ol style="list-style-type: none"> <li>For Unicode UTF8, set the environment variable to: <code>American_America.UTF8</code></li> <li>For Unicode AL32UTF8, set the environment variable to: <code>American_America.AL32UTF8</code></li> </ol>
<p>When starting pmsserver on UNIX, the following error message appears:</p> <pre>INFO: LM_36039 [Thu Mar 27 11:40:29 2003]: (6772   1) The maximum number of sessions that can run simultaneously is [10]. FATAL ERROR: CMN_1011 [Thu Mar 27 11:40:29 2003]: (6772   1) Error allocating system shared memory of [3000000] bytes for [Load Manager Shared Memory]. Error is [0]: [] FATAL ERROR: SF_34004 [Thu Mar 27 11:40:29 2003]: (6772   1) Server initialization failed. INFO: SF_34014 [Thu Mar 27 11:40:29 2003]: (6772   1) Server shut down.</pre>	<p>In the pmsserver.cfg file, change MaxSessions to 15 and change LMSharedMem to 3000000.</p> <p><b>Note:</b> For each 10 count increase of MaxSessions, increase the LMSharedMem by 2000000 bytes.</p>
<p>A session fails (any session), and the following error message appears:</p> <pre>TM_6227 Error: codepage incompatible in session [SDE_RecordExtractStart]. The source database DB2_OLTP and server ORACLE_BI_DW_SERVER do not have compatible code pages. (One way compatibility is required).</pre>	<p>For Informatica servers running on UNIX, in the pmsserver.cfg file, set ValidateDataCodePages to No.</p> <p>For Informatica servers running on Windows, navigate to Informatica Server setup, then Configuration, and remove the check from the Validate Data CodePages check box.</p>
<p>When trying to start the Informatica Server, the following error message appears:</p> <pre>sdc50a125{qauser7a}/&gt; pmsserver exec(): 0509-036 Cannot load program pmsserver because of the following errors: 0509-150 Dependent module libpmcef.a could not be loaded. 0509-022 Cannot load module libpmcef.a. 0509-026 System error: A file or directory in the path name does not exist.</pre>	<p>Set the environment parameters as follows:</p> <pre>setenv PATH /export/home/informatica/pm:~:\${PATH} setenv LD_LIBRARY_PATH /export/home/informatica/ pm:\${LD_LIBRARY_PATH}</pre>

## I.5 Oracle Business Analytics Warehouse Upgrade Issues

Table I-5 provides information about problems and solutions related to issues with upgrading the Oracle Business Analytics Warehouse.

**Table I-5 Upgrade Issues**

Symptom/Error Message	Probable Cause/Solution
When upgrading from Oracle Business Analytics Warehouse version 7.5.3 to version 7.7, duplicate rows are inserted into a mini dimension table from the dimension table.	<p>Before upgrading, all leading and trailing spaces need to be removed from the data.</p> <p>Before running the workflow for the mini dimension that appears in the upgrade folder, modify the data to remove leading and trailing spaces by using the RTRIM and LTRIM functions. For example:  <code>set (col)=rtrim(col)</code></p> <p>In cases where you find leading and trailing spaces, modify the SQL override for the mini dimension mapping in the main folder as shown in the example below:</p> <pre> SELECT DISTINCT     CASE WHEN G1.VAL IS NULL THEN RTRIM(LTRIM(DS.AREA)) ELSE G1.VAL END AS AREA,     CASE WHEN G1.NAME IS NULL THEN RTRIM(LTRIM(DS.AREA)) ELSE G1.NAME END AS AREA_I,     CASE WHEN G2.VAL IS NULL THEN RTRIM(LTRIM(DS.SUB_AREA)) ELSE G2.VAL END AS SUB_AREA     CASE WHEN G2.NAME IS NULL THEN RTRIM (LTRIM(DS.SUB_ AREA)) ELSE G2.NAME END AS SUB_AREA_I     CASE WHEN G3.VAL IS NULL THEN RTRIM(LTRIM(DS.SEV_CD)) ELSE G3.VAL END AS SEV_CD     CASE WHEN G3.NAME IS NULL THEN RTRIM (LTRIM(DS.SEV_CD)) ELSE G3.NAME END AS SEV_CD_I     CASE WHEN G4.VAL IS NULL THEN RTRIM(LTRIM(DS.PRIO_CD)) ELSE G4.VAL END AS PRIO_CD     CASE WHEN G4.NAME IS NULL THEN RTRIM (LTRIM(DS.PRIO_ CD)) ELSE G4.NAME END AS PRIO_CD_I FROM     W_SRVREQ_DS DS,     W_LOV_G G1,     W_LOV_G G2,     W_LOV_G G3,     W_LOV_G G4 WHERE     W_SRVREQ_DS LEFT OUTER JOIN V_LOV_G G1 ON     DS.AREA = G1.NAMEVAL AND     G1.TYPE = 'SR_AREA' LEFT OUTER JOIN W_LOV_G G2 ON     DS.SUB_AREA = G2.NAMEVAL AND     G2.TYPE = 'SR_AREA' LEFT OUTER JOIN W_LOV_G G3 ON     DS.SEV_CD = G3.NAMEVAL AND     G3.TYPE = 'SR_SEVERITY' LEFT OUTER JOIN W_LOV_G G4 ON     DS.PRIO_CD = G4.NAMEVAL AND     G4.TYPE = 'SR_PRIORITY'} </pre>

## I.6 Oracle BI Interactive Dashboard Issues

Table I-6 provides information about problems and solutions related to Oracle BI Interactive Dashboards.

**Table I-6 Oracle BI Dashboard Issues**

Symptom/Error Message	Probable Cause/Solution
Error Codes: OPR4ONWY:U9IM8TAC:OI2DL65P File: ./project/webodbcaccess/odbcstatement impl.cpp, Line: 186 State: S1000. Code: 10058. [NQODBC] [SQL_STATE: S1000] [nQSError: 10058] A general error has occurred. [nQSError: 17001] Oracle Error code: 4030, message: ORA-04030: out of process memory when trying to allocate 246284 bytes (hash-join subh,killcqas:killsltba) at OCI call OCISmtmFetch. [nQSError: 17012] Bulk fetch failed. (S1000)	This is caused by a shortage of RAM on a dedicated (non shared server) environment. Depending on the Oracle version, the options for DBAs are: <ul style="list-style-type: none"> <li>▪ Increase pga_aggregate_target.</li> <li>▪ Decrease sort_area_size and/or hash_area_size.</li> <li>▪ Move to multi-threaded server (a.k.a. MTS or shared servers).</li> </ul>

## I.7 ETL Issues

[Table I-7](#) provides information about problems and solutions related to ETL in Oracle BI Applications.

**Table I-7 ETL Issues**

Symptom/Error Message	Probable Cause/Solution
ETL failure with error: unable to create unique indexes over the W_UAEMP_ TMP and W_UAPOS_TMP tables.	Duplicate records exist in the WS_PER_RESP table, probably caused by Responsibility changes in the Siebel CRM Administration application.  For example, in Siebel CRM Administration View 'Responsibilities', you remove a Responsibility for a particular User, and later re-associate that Responsibility to that User.



---

---

## About the Versioned Informatica Repository

This section explains the versioned Informatica Repository that is included in Oracle BI Applications, and contains the following topics:

- [Section J.1, "Summary of Versioning in the Informatica Repository"](#)
- [Section J.2, "Customization of Repository Objects Using Check Out and Check In"](#)

### J.1 Summary of Versioning in the Informatica Repository

Oracle BI Applications 7.9.5 ships the following two Informatica Repository files:

- Oracle\_BI\_DW\_Base.rep
- Oracle\_BI\_DW\_Teradata.rep

In releases 7.9.0, 7.9.1 and 7.9.2, these were non-versioned files provided in the standard application. In Oracle BI Applications version 7.9.3 and later, these Informatica Repository files are now versioned. Because of this change, you need the Team based Development option. This option is included in the OEM license provided to you by Oracle. Also, if you extend or modify the standard mappings, you need to check out and check changes to the Informatica Repository. DAC will now only pick up and execute valid and checked-in objects from the Informatica Repository.

Oracle BI Applications now uses a standard way of labeling objects. For example, the standard Informatica objects have the label 'Oracle Business Intelligence Applications Release 7.9.x'. This label is useful when tracing back and comparing changes made from release to release. Note that Informatica does not allow a versioned repository file to be restored as a non-versioned repository somewhere else. Therefore, this change is uni-directional.

### J.2 Customization of Repository Objects Using Check Out and Check In

When working with an Informatica Repository that is versioned, you must check out a repository object to modify and then check in the change. To extend or modify the standard mappings, you must check out a repository object that is to be modified, make the desired changes, and then check in the changes. This process is described below, and is illustrated in the screenshot below where the red arrow represents the action flow.

The workflow for checking out and checking in changes in the Informatica Repository is described below:

- Developers check out an object for modification.

Note the green arrow on the object in the screenshot below.

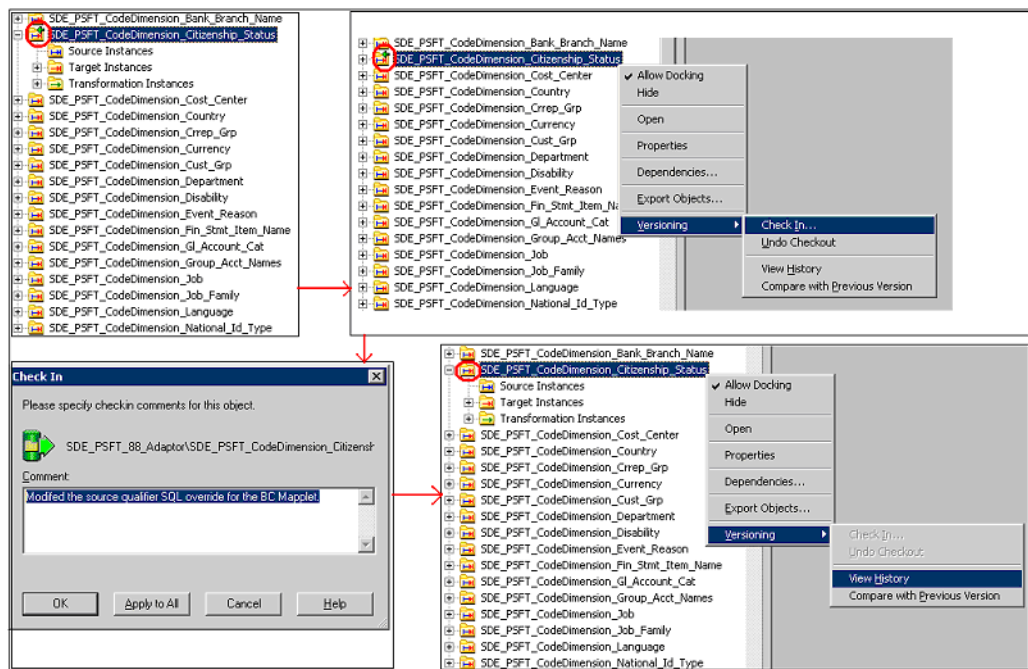
- Developers make changes, validate and save the object, and are ready to check their changes in.

The green arrow is still displayed to indicate this is still a checked out object.

- Developers provide mandatory check in comments and finally check in.
- The object is checked in now.

Note that the green arrow disappears to indicate that the current version is now checked in and is usable/visible by other developers (for example, in DAC). Developers can now also compare versions and view history for this object.

The version prior to this change also gets stored in the repository for tracking purposes. Apart from tracking changes in repository objects, Informatica provides additional useful features like Query and labeling on versioned repositories. Refer to the Informatica product guides for the more information about working with versioned objects.



---

---

# Index

## A

---

action links  
  about, D-6  
  testing, D-6  
alignments dimension, configuring, E-10  
ANSI NULL option, setting, 3-7

## B

---

browsers  
  settings, configuring, D-7  
bulkload mode, error while creating a custom session, I-4

## C

---

Catalog Manager  
  using, 2-5  
charts  
  local seed data, importing, C-2  
  translation tables, process of maintaining, C-1  
  W\_LOCALIZED\_STRING\_G table, adding custom translations, C-4  
  Web catalog strings, about translating, C-5  
circular hierarchy exceptions, about and example, H-3  
code page  
  environments, 4-6  
Communications, Media, and Energy Analytics  
  dashboards and pages, E-9  
  data requirements, E-10  
  order management analytics, about, E-7  
  predictive scores, developing and deploying, E-4  
complex products defined, E-7  
configuring, E-11  
  DAC client, 4-14  
  DAC metadata repository, A-9  
  email recipients in DAC client, 4-49  
  Oracle Business Analytics Warehouse database, 4-33  
  SSE role, 4-9  
Contact Center Telephony Analytics  
  dashboards and pages, E-5  
  data requirements, E-7  
Cost List exceptions

*See also* exception reports  
  about and example, H-2  
  Cost lists, using to display, H-6  
  products without a cost list, H-3  
CRM and EBS schema changes  
  verifying, 4-66  
currency, diagnosing translation issues, H-5

## D

---

DAC  
  configuring metadata repository, A-9  
  Source System Parameters, 4-61  
  starting and stopping the server, A-4  
DAC metadata repository  
  database, 4-8  
DAC Server  
  starting and stopping manually, A-4  
dashboards  
  CME dashboards and pages, E-9  
  consumer sector dashboards and pages, E-5  
  customizing, about, D-8  
  exposing dashboards and requests, D-14  
  information, about viewing information, D-14  
  locating dashboard and page names, D-14  
  viewing a list of, 2-5  
  views, adding, D-11  
data import  
  syndicated data loading, overview, F-2  
data loading  
  Oracle's Siebel Analytics for Life Sciences, issues when loading, F-30  
  troubleshooting, Informatica and data loading issues, I-2  
Data Movement Mode, 4-6  
data movement mode, 4-6  
data warehouse  
  database connections, 4-51  
Data Warehouse Administration Console (DAC)  
  configuring on UNIX, A-8  
  configuring the client, 4-14  
  setting up system properties, A-9  
database connections  
  transactional, creating, 4-51  
database requirements, 4-8  
DataMovementMode parameter, 4-6

- DB Library Options setting, modifying, 3-7
- DB2 databases
  - DB2CODEPAGE, 4-11
  - environment variable, 4-11
  - Oracle Business Analytics Warehouse requirements, 3-4
  - Oracle Business Analytics Warehouse, cannot connect to, I-2

- DB2-UDB
  - creating stored procedures, 4-62
- deployment
  - topology example, 4-2
- dimensions
  - alignments dimension, configuring, E-10
  - pharmaceuticals, configuring dimensions specific to, E-12
  - positions dimension, configuring, E-10
  - product costs, configuring, E-11
  - products dimension, configuring, E-11
- documentation
  - Repository Documentation feature, 2-4

## E

---

- email
  - configuring Email in the DAC Server, 4-60
  - notification in Outlook error when sending post session email, I-4
- environment variables
  - DB2CODEPAGE, 4-11
  - NLS\_LANG, 4-10
- ETL
  - error with W\_UAEMP\_TMP and W\_UAPOS\_TMP table, I-10
  - History view, H-6
  - improve performance using partitioning, 3-21
  - index errors, I-10
  - running with Non-English OLTP Data Source, 3-20
  - troubleshooting, I-10
- ETL failure, I-10
- exception reports
  - additional exceptions, list of, H-6
  - circular hierarchy exceptions, about and example, H-3
  - cleansing the data, process and diagram, H-4
  - ETL History view, using to view the history of the ETL processes and statuses, H-6
  - exception report components, list of, H-1
  - exception, fixing, H-4
  - exchange rate exceptions, about, H-3
  - Exchange Rate view, using to diagnose currency translation issues, H-5
  - executing the report, cleansing steps, H-4
  - invalid hierarchy exceptions, about, H-3
  - List of Values exceptions, H-1
  - List of Values view, using to compare values with Oracle Business Analytics Warehouse, H-5
  - See also* Cost List exceptions
- system preferences

- exchange rate exceptions, about, H-3
- Exchange Rate view, using to diagnose currency issues, H-5

## H

---

- hierarchical relationships, invalid hierarchy exceptions, about, H-3
- home page
  - content, customizing, D-8
  - customizing, about, D-8
  - path argument, determining, D-10
- Horizontal applications
  - creating indexes for performance, 3-16

## I

---

- Incentive Compensation Analytics, turning on, D-7
- indexes
  - creating to improve performance, 3-16
  - S\_\* tables, 3-17
- Informatica
  - default passwords, 4-18
  - fatal errors 11, I-6
  - fatal errors 3644-1 or 360454, I-6
- Informatica PowerCenter
  - requirements, 4-6
  - terminology changes in 8.1.1, 4-6
- Informatica repository
  - database, 4-8
  - dropping, A-7
  - versioned, J-1
- Informatica Server
  - password, 4-18
  - registering, 4-49
- infrastructure requirements, 4-5
- Initialization Blocks, 3-16
- installation
  - database connectivity, 4-26
  - Java SDK, 4-14
  - JDBC drivers, 4-26
  - Oracle BI Applications installation process task list, 4-3
- installation and configuration
  - task list, 4-3
- IP address
  - virtual IP address, creating, D-5
  - virtual IP address, testing, D-5

## J

---

- Java JDK
  - requirements, 4-14
- Java SDK
  - installation, 4-14
- JDBC drivers
  - installation, 4-26

## L

---

- large fact tables

- partitioning, 3-21
- List of Values
  - exceptions, about and example, H-1
  - view, using to compare values with the Oracle Business Analytics Warehouse, H-5

## M

---

- Mappings
  - SA System Subject Area, 3-15
- metadata
  - externalizing strings, C-3
- Microsoft Outlook
  - closes when finishing ETL process, probable cause, I-4
  - error when sending post session email, I-4
- Migrating Seed Data from Siebel OLTP Source Systems, 4-63

## N

---

- NLS\_LANG environment variable, 4-10
- Non-English OLTP Data Source
  - running an ETL with, 3-20

## O

---

- OPB\_OBJECT\_LOCKS Informatica session
  - deadlocked, probable cause, I-4
- Operational Applications
  - dashboards, adding views, D-11
  - home page content, customizing, D-8
  - home page path argument, determining, D-10
- Oracle
  - mappings hang, probable cause, I-3
- Oracle BI Administration Tool
  - using to generate Repository Documentation, 2-4
- Oracle BI Applications
  - action links, about, D-6
  - action links, testing, D-6
  - browser settings, configuring, D-7
  - completing initialization, D-2
  - dashboards, adding views, D-11
  - home page and dashboards, about customizing, D-8
  - home page content, customizing, D-8
  - Incentive Compensation Analytics, turning on, D-7
  - initialization, completing, D-2
  - installation and configuration process task list, 4-3
  - operational applications, about integrating within, D-6
  - Optional Analytics Applications, D-7
  - virtual IP address, creating, D-5
  - Web Extension, configuring to work with, D-4
- Oracle BI Applications, configuring
  - home page, determining path argument, D-10
  - virtual IP address, testing, D-5
- Oracle BI Charts
  - metadata strings, externalizing, C-3

- Oracle BI seed data, updating for non-English locales, C-2
- Oracle BI EE
  - requirements, 4-5
- Oracle BI Presentation layer
  - process of maintaining translation tables, C-1
- Oracle BI repository
  - externalizing metadata strings, C-3
- Oracle BI SE One
  - requirements, 4-5
- Oracle BI seed data
  - updating for non-English locales, C-2
- Oracle Business Analytics Warehouse
  - architecture components, 2-2
  - database, 4-8
  - database parameters and requirements, 3-2
  - database, configuring, 4-33
  - DB2 requirements, 3-4
  - operating system, driver, connectivity requirements, 3-2
  - Oracle requirements, 3-13
  - SQL server-specific requirements, 3-6
  - tables, creating, 4-34
  - Teradata requirements, 3-9
- Oracle databases
  - NLS\_LANG, 4-10
- Oracle Financial Analytics
  - disabling logical table sources, E-3
  - externalizing metadata translation strings, E-3
  - logical table sources, disabling, E-3
  - logical table sources, updating, E-2
  - metadata translation strings, externalizing, E-3
  - updating logical table sources, E-2
- Oracle's Siebel Life Sciences
  - data loading issues, F-30
- Oracle's Siebel Industry Applications
  - alignments dimension, configuring, E-10
  - Analytics applications, about metadata requirements, E-5
  - CME dashboards and CME Analytics applications dashboards and pages, E-9
  - CME data requirements, E-10
  - consumer sector dashboards and pages, E-5
  - consumer sector data requirements, E-7
  - dimensions, configuring specific to pharmaceuticals, E-12
  - Pharma sales effectiveness, configuring subject area, E-12
  - product costs dimension, configuring, E-11
- Oracle's Siebel Industry applications
  - alignments dimension, configuring, E-10
  - Analytics applications, about metadata requirements, E-5
  - CME dashboards and pages, E-9
  - CME data requirements, E-10
  - consumer sector dashboards and pages, E-5
  - consumer sector data requirements, E-7
  - dimensions, configuring specific to pharmaceuticals, E-12
  - Pharma medical education effectiveness,

- configuring subject area, E-13
- Pharma product categories, configuring subject area, E-12
- Pharma promotional effectiveness, configuring subject area, E-12
- Pharma Sales Analytics, about metadata requirements, E-10
- Pharma sales effectiveness, configuring subject area, E-12
- positions dimension, configuring, E-10
- predictive scores, developing and deploying, E-4
- product costs dimension, configuring, E-11
- Oracle's Siebel Operational Applications integration with
  - data types, D-1
  - integration with Industry Applications, D-1
- order management analytics, about, E-7
- Outlook
  - closes when finishing ETL process, probable cause, I-4
  - error when sending post session email, I-4

## P

---

- page names, locating and dashboards, D-14
- parent-child relationship, circular hierarchy exceptions, H-3
- partitioning, 3-21
- password
  - Informatica administrator, 4-18
- PerfIndex\_Horizontal.sql, 3-17
- PerfIndex\_Industry.sql, 3-17
- performance
  - components configured to optimize performance, 3-7
  - implementing indexes on Oracle sources to improve, 3-17
  - implementing indexes on Siebel sources to improve, 3-16
  - partitioning large fact tables, 3-21
- Pharma objectives achievement
  - configuring subject area, E-13
  - medical education effectiveness, configuring subject area, E-13
- Pharma product categories
  - configuring subject area, E-12
- Pharma promotional effectiveness, configuring subject area, E-12
- Pharma Sales Analytics
  - alignments dimension, configuring, E-10
  - dimensions, configuring specific to pharmaceuticals, E-12
  - medical education effectiveness, configuring subject area, E-13
  - metadata requirements, about, E-10
  - objectives achievement, configuring subject area, E-13
  - Pharma medical education effectiveness, configuring subject area, E-13
  - Pharma objectives achievement, configuring

- subject area, E-13
- Pharma product categories, configuring subject area, E-12
- Pharma promotional effectiveness, configuring subject area, E-12
- Pharma Sales effectiveness, configuring subject area, E-12
- positions dimension, configuring, E-10
- product categories, configuring subject area, E-12
- product costs dimension, configuring, E-11
- products dimension, configuring, E-11
- promotional effectiveness, configuring subject area, E-12
- sales effectiveness, configuring subject area, E-12
- Pharma Sales Analytics applications
  - dimensions, configuring specific to pharmaceuticals, E-12
  - metadata requirements, about, E-10
- Pharma medical education effectiveness, configuring subject area, E-13
- Pharma objectives achievement, configuring subject area, E-13
- Pharma product categories, configuring subject area, E-12
- Pharma promotional effectiveness, configuring subject area, E-12
- Pharma Sales effectiveness, configuring subject area, E-12
- positions dimension, configuring, E-10
- product costs dimension, configuring, E-11
- products dimension, configuring, E-11
- Pharma Sales Analytics, about metadata requirements, E-10
- positions dimension, configuring, E-10
- predictive analytics metadata
  - predictive scores, developing and deploying, E-4
- predictive scores, developing and deploying, E-4
- Presentation Catalog, administering
  - dashboards, about viewing information, D-14
  - dashboards, exposing and requests, D-14
  - dashboards, locating and page names, D-14
- Presentation Services Catalog
  - viewing, 2-5
- product costs dimension, configuring, E-11
- products dimension, configuring, E-11

## Q

---

- Query, I-2
- Query Analyzer, can not open after installing ODBC driver, I-2
- query performance
  - improve using partitioning, 3-21

## R

---

- RDBMS error message, probable cause, I-3
- repository
  - versioned, J-1
- Repository Documentation

- generating, 2-4
- requests
  - exposing dashboards and requests, D-14
- requirements
  - database, 4-8
  - Informatica PowerCenter, 4-6
  - infrastructure, 4-5
  - Java SDK, 4-14
  - Oracle BI EE, 4-5
  - Oracle BI SE One, 4-5
- responsibilities
  - in Usage Accelerator, I-10

## S

---

- S\_.\* tables
  - indexes on, 3-17
- SA System Subject Area
  - mapping, 3-15
- schema definitions
  - transactional database, updating, 4-64
- SDEINC\_RecordExtractStart session fails, probable cause, I-3
- security
  - initialization blocks, 3-16
- seed data
  - importing, C-2
- seed data, importing, D-1
- server components, installation requirements
  - See also* server component software
- setting up
  - DAC system properties, A-9
- simple products defined, E-8
- Source System Parameters
  - about, 4-61
  - setting, 4-61
- Source System Parameters (also known as DAC ETL Preferences), 4-61
- SQL Server
  - DB Library Options setting, modifying, 3-7
  - Oracle Business Analytics Warehouse requirements, 3-6
- SSE role
  - creating, 4-9
  - requirements, 4-9
- starting the DAC Server, A-4
- stored procedures for DB2-UDB
  - creating, 4-62
- string translations
  - adding for metadata, C-4
- SWE
  - See* Web Extension
- syndicated data
  - importing, overview, F-2
- system preferences
  - See also* exception reports

## T

---

- Task List

- installation and configuration, 4-3
- Teradata
  - activating join indexes, A-13
  - avoiding arithmetic overflows, 3-9
  - code page file property, setting, 4-67
  - configuration checklist, Windows, 4-68
  - creating and dropping tables, 4-37
  - data warehouse requirements, 3-9
  - database hints, 3-9
  - general guidelines, 3-9
  - Hosts file, setting up, 4-67
- terminology
  - changes in Informatica PowerCenter 8.1.1, 4-6
- topology
  - deploying Oracle BI Applications, 4-2
  - example, 4-2
- Transactional database
  - connection error, I-2
  - Cost lists, displaying, H-6
  - importing locale seed data, C-2
- translation tables
  - process of maintaining, C-1
- troubleshooting
  - ETL error, I-6, I-10
  - ETL failure, I-10
  - general, I-1
  - Informatica and data loading issues, I-2
  - Oracle Business Analytics Warehouse
    - installation, I-2
  - Unicode, I-6, I-9, I-10
  - UNIX installation, I-7
  - W\_UAEMP\_TMP, I-10
  - W\_UAPOS\_TMP, I-10

## U

---

- Unicode
  - local seed data, importing, C-2
  - metadata strings, externalizing, C-3
  - Oracle BI seed data, updating for non-English locales, C-2
  - string translations, adding for metadata, C-4
  - translation tables, process of maintaining, C-1
  - W\_LOCALIZED\_STRING\_G table, adding custom translations, C-4
  - Web catalog strings, about translating, C-5
- Universal Adapter
  - for CRM Analytics, B-1
- URL
  - dashboards, configuring for, D-11
  - home page, configuring for, D-8
- Usage Accelerator
  - ETL failure, I-10

## V

---

- verifying schema changes in Oracle's Siebel (CRM) and EBS, 4-66
- versioned Informatica repository, J-1
- versioned objects

- Informatica repository, J-1
- Vertical applications
  - creating indexes for performance, 3-16
- views, adding to dashboards, D-11
- virtual IP address
  - creating, D-5
  - testing, D-5

## **W**

---

- W\_LOCALIZED\_STRING\_G table, adding custom translations, C-4
- W\_UAEMP\_TMP
  - ETL errors, I-10
- W\_UAPOS\_TMP
  - ETL errors, I-10
- Web Catalog strings
  - about translating, C-5
- Web Extension
  - configuring to work with, D-4
  - virtual IP address, creating, D-5
  - virtual IP address, testing, D-5
- workflows
  - error message when executing on the Informatica Server Manager, probable cause, I-3
  - Server Manager error message, probable cause, I-2