

Oracle® Business Intelligence Applications

Upgrade Guide for Informatica PowerCenter Users

Release 7.9.6.3

E19040-01

April 2011

Copyright © 2001, 2011, Oracle and/or its affiliates. All rights reserved.

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

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

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, 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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	xv
Audience	xvi
Documentation Accessibility	xvi
Related Documents	xvii
Conventions	xvii
1 What's New in This Release	
1.1 What's New in Oracle BI Applications Release 7.9.6.3	1-1
2 Overview of Oracle BI Applications Upgrade	
2.1 Supported Upgrade Paths	2-1
2.2 Why Upgrade to Oracle BI Applications Release 7.9.6.3?	2-2
2.2.1 Upgrade Options	2-2
2.3 Considerations About Upgrading to Oracle BI Enterprise Edition 11g	2-3
2.4 Considerations About Upgrading Oracle BI Applications	2-3
2.5 Best Practices for Preparing to Upgrade	2-5
2.6 Major Stages of Oracle BI Applications Upgrade	2-6
2.6.1 Sequence of Upgrade Stages	2-6
2.6.2 Documentation Related to the Major Stages of the Upgrade Process	2-9
Part I Upgrading When Your Source System is Siebel CRM	
3 Upgrading Siebel Analytics 7.5.x	
3.1 Upgrading Oracle BI Infrastructure	3-2
3.2 Upgrading Oracle BI Applications	3-2
3.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2	3-3
3.4 Upgrading the Informatica Repository	3-3
3.5 Configuring Informatica PowerCenter Version 9.0.1 to Work With Oracle BI Applications and DAC	3-4
3.6 Installing and Configuring DAC	3-5
3.7 Overview of Upgrading the Data Warehouse Schema and Migrating Data	3-5
3.8 Upgrading the Data Warehouse Schema to Version 7.7 and Migrating Data	3-6
3.9 Upgrading the Data Warehouse Schema to Version 7.9.0	3-9
3.10 Importing New Schema Definitions into the Siebel Transactional Database	3-10
3.10.1 Verifying the Siebel Transactional Database Upgrade	3-10

3.11	Migrating Data into the Data Warehouse Upgraded to Version 7.9.0.....	3-11
3.12	Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data	3-14
3.13	Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data	3-15
3.14	Upgrading the Data Warehouse Schema to Version 7.9.5.1	3-16
3.15	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data	3-16
3.15.1	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1	3-17
3.15.2	Migrating Data Into the Upgraded Data Warehouse.....	3-18
3.15.3	Verifying the Data Migrated Successfully	3-19
3.16	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1.....	3-20
3.16.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1	3-20
3.16.2	Migrating Data into the Upgraded Data Warehouse	3-20
3.16.3	Verifying the Data Migrated Successfully	3-21
3.17	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data	3-21
3.17.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	3-21
3.17.2	Migrating Data into the Upgraded Data Warehouse	3-22
3.17.3	Verifying the Data Migrated Successfully	3-22
3.18	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data	3-23
3.18.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	3-23
3.18.2	Migrating Data Into the Upgraded Data Warehouse.....	3-24
3.18.3	Verifying the Data Migrated Successfully	3-25
3.19	Localizing the Policy Store	3-26
3.20	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	3-26
3.21	Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version	3-26
3.22	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	3-27
3.22.1	Preparing for the Oracle BI Repository Upgrade.....	3-28
3.22.2	Equalizing the Oracle BI Repositories	3-29
3.22.3	Comparing the Oracle BI Repositories	3-31
3.22.4	Merging the Oracle BI Repositories	3-31
3.23	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	3-32
3.23.1	Trimming the Input Presentation Catalog	3-33
3.23.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	3-33
3.23.3	Testing the Results of the Presentation Catalog Upgrade	3-34
3.24	Regression Testing the Oracle BI Repository Merge	3-35

4 Upgrading Siebel Business Analytics 7.7, 7.7.x and 7.8.x

4.1	Upgrading Oracle BI Infrastructure	4-2
4.2	Upgrading Oracle BI Applications.....	4-3
4.3	Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2.....	4-3
4.4	Upgrading the Informatica Repository.....	4-4

4.5	Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC	4-5
4.6	Upgrading and Configuring DAC	4-5
4.6.1	Installing the DAC Platform and Oracle BI Applications Metadata Repository Files.....	4-5
4.6.2	Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica	4-6
4.6.3	Upgrading the DAC Repository for Siebel Business Analytics	4-6
4.7	Overview of Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases	4-9
4.8	Upgrading the Data Warehouse Schema to Version 7.9.0 for Non-Teradata Databases.....	4-10
4.9	Importing New Schema Definitions Into the Siebel Transactional Database	4-11
4.9.1	Verifying the Siebel Transactional Database Upgrade.....	4-11
4.10	Migrating Data into the Data Warehouse Upgraded to Version 7.9.0 for Non-Teradata Databases	4-12
4.11	Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data for Non-Teradata Databases	4-16
4.12	Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data for Non-Teradata Databases	4-17
4.13	Upgrading the Data Warehouse Schema to Version 7.9.5.1 for Non-Teradata Databases.....	4-18
4.14	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data for Non-Teradata Databases	4-19
4.14.1	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1	4-19
4.14.2	Migrating Data Into the Upgraded Data Warehouse.....	4-20
4.14.3	Verifying the Data Migrated Successfully	4-21
4.15	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases	4-22
4.15.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1	4-22
4.15.2	Migrating Data into the Upgraded Data Warehouse.....	4-22
4.15.3	Verifying the Data Migrated Successfully	4-23
4.16	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases	4-23
4.16.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	4-24
4.16.2	Migrating Data into the Upgraded Data Warehouse.....	4-24
4.16.3	Verifying the Data Migrated Successfully	4-24
4.17	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases	4-25
4.17.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	4-26
4.17.2	Migrating Data Into the Upgraded Data Warehouse.....	4-26
4.17.3	Verifying the Data Migrated Successfully	4-27
4.18	Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases	4-28
4.18.1	Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3.....	4-28
4.18.2	Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3.....	4-34
4.18.3	Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3.....	4-40
4.18.4	Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3.....	4-45
4.18.5	Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3.....	4-50

4.18.6	Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3	4-53
4.19	Localizing the Policy Store	4-56
4.20	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	4-56
4.21	Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version	4-57
4.22	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	4-58
4.22.1	Preparing for the Oracle BI Repository Upgrade.....	4-58
4.22.2	Equalizing the Oracle BI Repositories	4-59
4.22.3	Comparing the Oracle BI Repositories	4-61
4.22.4	Merging the Oracle BI Repositories	4-61
4.23	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	4-63
4.23.1	Trimming the Input Presentation Catalog.....	4-63
4.23.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	4-64
4.23.3	Testing the Results of the Presentation Catalog Upgrade	4-65
4.24	Regression Testing the Oracle BI Repository Merge	4-65

5 Upgrading Oracle BI Applications 7.9.x for Siebel Source Systems

5.1	Upgrading Oracle BI Infrastructure	5-2
5.2	Upgrading Oracle BI Applications.....	5-2
5.3	Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2.....	5-2
5.4	Upgrading the Informatica Repository.....	5-3
5.5	Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC.....	5-4
5.6	Upgrading and Configuring DAC	5-4
5.6.1	Installing the DAC Platform and Oracle BI Applications Metadata Repository Files.....	5-5
5.6.2	Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica	5-5
5.6.3	Upgrading the DAC Repository.....	5-5
5.7	Importing New Schema Definitions Into the Siebel Transactional Database	5-6
5.7.1	Verifying the Siebel Transactional Database Upgrade.....	5-7
5.8	Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases.....	5-7
5.8.1	Upgrading the Data Warehouse Schema From Version 7.9.0 to 7.9.4 and Migrating Data	5-8
5.8.2	Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data	5-9
5.8.3	Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data	5-10
5.8.4	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data	5-10
5.8.4.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	5-10
5.8.4.2	Migrating Data Into the Upgraded Data Warehouse.....	5-12
5.8.4.3	Verifying the Data Migrated Successfully	5-13

5.8.5	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data	5-13
5.8.5.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	5-14
5.8.5.2	Migrating Data Into the Upgraded Data Warehouse.....	5-14
5.8.5.3	Verifying the Data Migrated Successfully	5-14
5.8.6	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data	5-15
5.8.6.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	5-15
5.8.6.2	Migrating Data Into the Upgraded Data Warehouse.....	5-16
5.8.6.3	Verifying the Data Migrated Successfully	5-16
5.8.7	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data	5-17
5.8.7.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	5-17
5.8.7.2	Migrating Data Into the Upgraded Data Warehouse.....	5-18
5.8.7.3	Verifying the Data Migrated Successfully	5-19
5.9	Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases	5-20
5.9.1	Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3.....	5-20
5.9.2	Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3.....	5-26
5.9.3	Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3.....	5-32
5.9.4	Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3.....	5-37
5.9.5	Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3.....	5-42
5.9.6	Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3.....	5-45
5.10	Localizing the Policy Store	5-47
5.11	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	5-48
5.12	Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version	5-48
5.13	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	5-49
5.13.1	Preparing for the Oracle BI Repository Upgrade.....	5-50
5.13.2	Equalizing the Oracle BI Repositories	5-50
5.13.3	Comparing the Oracle BI Repositories	5-53
5.13.4	Merging the Oracle BI Repositories	5-53
5.14	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	5-54
5.14.1	Trimming the Input Presentation Catalog.....	5-55
5.14.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	5-55
5.14.3	Testing the Results of the Presentation Catalog Upgrade	5-56
5.15	Regression Testing the Oracle BI Repository Merge	5-57

Part II Upgrading When Your Source System Is Oracle EBS

6 Upgrading Oracle BI Applications for Oracle Source Systems

6.1	Upgrading Oracle BI Infrastructure	6-2
6.2	Upgrading Oracle BI Applications	6-2
6.3	Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2.....	6-2

6.4	Upgrading the Informatica Repository.....	6-3
6.5	Configuring Informatica PowerCenter Version 9.0.1 Hotfix 2 to Work With Oracle BI Applications and DAC	6-4
6.6	Upgrading and Configuring DAC	6-4
6.6.1	Installing the DAC Platform and Oracle BI Applications Metadata Repository Files.....	6-4
6.6.2	Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica	6-5
6.6.3	Upgrading the DAC Repository.....	6-5
6.7	Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases.....	6-6
6.7.1	Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases.....	6-7
6.7.2	Upgrading the Data Warehouse Schema From Version 7.9.x to 7.9.3 and Migrating Data	6-7
6.7.3	Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data	6-8
6.7.4	Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5. and Migrating Data	6-10
6.7.5	Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data	6-11
6.7.6	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data	6-12
6.7.6.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	6-12
6.7.6.2	Migrating Data into the Upgraded Data Warehouse.....	6-13
6.7.6.3	Verifying the Data Migrated Successfully	6-15
6.7.6.4	Resetting Refresh Dates	6-15
6.7.7	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data	6-16
6.7.7.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	6-16
6.7.7.2	Migrating Data into the Upgraded Data Warehouse.....	6-16
6.7.7.3	Verifying the Data Migrated Successfully	6-17
6.7.8	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data	6-18
6.7.8.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	6-18
6.7.8.2	Migrating Data into the Upgraded Data Warehouse.....	6-18
6.7.8.3	Verifying the Data Migrated Successfully	6-19
6.7.9	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data	6-20
6.7.9.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	6-20
6.7.9.2	Migrating Data into the Upgraded Data Warehouse.....	6-21
6.7.9.3	Verifying the Data Migrated Successfully	6-22
6.8	Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases	6-23
6.8.1	Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3.....	6-23
6.8.2	Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3.....	6-32
6.8.3	Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3.....	6-39
6.8.4	Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3.....	6-46
6.8.5	Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3.....	6-53
6.8.6	Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3.....	6-58

6.8.7	Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3	6-62
6.9	Localizing the Policy Store	6-65
6.10	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	6-65
6.11	Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version	6-66
6.12	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	6-66
6.12.1	Preparing for the Oracle BI Repository Upgrade.....	6-67
6.12.2	Equalizing the Oracle BI Repositories	6-68
6.12.3	Comparing the Oracle BI Repositories	6-70
6.12.4	Merging the Oracle BI Repositories	6-70
6.13	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	6-72
6.13.1	Trimming the Input Presentation Catalog.....	6-72
6.13.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	6-73
6.13.3	Testing the Results of the Presentation Catalog Upgrade	6-74
6.14	Regression Testing the Oracle BI Repository Merge	6-74

Part III Upgrading When Your Source System is PeopleSoft

7 Upgrading Oracle BI Applications for PeopleSoft Source Systems

7.1	Upgrading Oracle BI Infrastructure	7-2
7.2	Upgrading Oracle BI Applications.....	7-2
7.3	Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2.....	7-2
7.4	Upgrading the Informatica Repository.....	7-3
7.5	Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC	7-4
7.6	Upgrading and Configuring DAC	7-4
7.6.1	Installing the DAC Platform and Oracle BI Applications Metadata Repository Files.....	7-4
7.6.2	Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica	7-5
7.6.3	Upgrading the DAC Repository.....	7-5
7.7	Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases.....	7-6
7.7.1	Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases	7-7
7.7.2	Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data	7-8
7.7.3	Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data	7-9
7.7.4	Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data	7-10
7.7.5	Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data	7-11
7.7.5.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	7-11

7.7.5.2	Migrating Data into the Upgraded Data Warehouse.....	7-12
7.7.5.3	Verifying the Data Migrated Successfully	7-13
7.7.6	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data	7-14
7.7.6.1	Upgrading the Data Warehouse Schema to Version 7.9.6.1.....	7-14
7.7.6.2	Migrating Data into the Upgraded Data Warehouse.....	7-15
7.7.6.3	Verifying the Data Migrated Successfully	7-15
7.7.7	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data	7-16
7.7.7.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	7-16
7.7.7.2	Migrating Data into the Upgraded Data Warehouse.....	7-17
7.7.7.3	Verifying the Data Migrated Successfully	7-17
7.7.8	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data	7-18
7.7.8.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	7-19
7.7.8.2	Migrating Data into the Upgraded Data Warehouse.....	7-19
7.7.8.3	Verifying the Data Migrated Successfully	7-20
7.8	Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases	7-21
7.8.1	Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3.....	7-21
7.8.2	Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3.....	7-28
7.8.3	Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3.....	7-35
7.8.4	Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3.....	7-41
7.8.5	Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3.....	7-46
7.8.6	Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3.....	7-49
7.9	Localizing the Policy Store	7-52
7.10	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	7-53
7.11	Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version	7-53
7.12	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	7-54
7.12.1	Preparing for the Oracle BI Repository Upgrade.....	7-55
7.12.2	Equalizing the Oracle BI Repositories	7-55
7.12.3	Comparing the Oracle BI Repositories	7-58
7.12.4	Merging the Oracle BI Repositories	7-58
7.13	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	7-59
7.13.1	Trimming the Input Presentation Catalog.....	7-60
7.13.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog.....	7-60
7.13.3	Testing the Results of the Presentation Catalog Upgrade	7-61
7.14	Regression Testing the Oracle BI Repository Merge	7-62

Part IV Upgrading When Your Source System is JD Edwards

8 Upgrading Oracle BI Applications for JD Edwards Source Systems

8.1	Upgrading Oracle BI Infrastructure	8-2
8.2	Upgrading Oracle BI Applications.....	8-2

8.3	Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2.....	8-2
8.4	Upgrading the Informatica Repository.....	8-3
8.5	Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC	8-4
8.6	Upgrading and Configuring DAC	8-4
8.6.1	Installing the DAC Platform and Oracle BI Applications Metadata Repository Files.....	8-4
8.6.2	Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica	8-5
8.6.3	Upgrading the DAC Repository.....	8-5
8.7	Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases.....	8-6
8.7.1	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases	8-6
8.7.1.1	Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1	8-6
8.7.1.2	Migrating Data into the Upgraded Data Warehouse.....	8-7
8.7.1.3	Verifying the Data Migrated Successfully	8-7
8.7.2	Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases	8-8
8.7.2.1	Upgrading the Data Warehouse Schema to Version 7.9.6.2.....	8-8
8.7.2.2	Migrating Data into the Upgraded Data Warehouse.....	8-8
8.7.2.3	Verifying the Data Migrated Successfully	8-9
8.7.2.4	Resetting Refresh Dates	8-10
8.7.3	Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases	8-10
8.7.3.1	Upgrading the Data Warehouse Schema to Version 7.9.6.3.....	8-11
8.7.3.2	Migrating Data into the Upgraded Data Warehouse.....	8-11
8.7.3.3	Verifying the Data Migrated Successfully	8-12
8.8	Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases	8-13
8.8.1	Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.2.....	8-13
8.8.2	Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3.....	8-17
8.9	Localizing the Policy Store	8-20
8.10	Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server	8-20
8.11	Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version	8-20
8.12	Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository	8-21
8.12.1	Preparing for the Oracle BI Repository Upgrade.....	8-22
8.12.2	Equalizing the Oracle BI Repositories	8-23
8.12.3	Comparing the Oracle BI Repositories	8-25
8.12.4	Merging the Oracle BI Repositories	8-25
8.13	Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	8-26
8.13.1	Trimming the Input Presentation Catalog	8-27
8.13.2	Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog	8-27
8.13.3	Testing the Results of the Presentation Catalog Upgrade	8-28

Part V Appendices

A Setting Parameters in Informatica Parameter Files

A.1	Setting Common Parameters in Informatica Parameter Files	A-1
A.1.1	Setting Parameters in 793_UPG_PARAMS.txt	A-1
A.1.2	Setting Parameters in 794_UPG_PARAMS.txt	A-2
A.1.3	Setting Parameters in 795_UPG_PARAMS.txt	A-2
A.1.4	Setting Parameters in 7951_UPG_PARAMS.txt	A-3
A.1.5	Setting Parameters in 796_UPG_PARAMS.txt	A-3
A.1.6	Setting Parameters in 7961_UPG_PARAMS.txt	A-4
A.1.7	Setting Parameters in 7962_UPG_PARAMS.txt	A-5
A.1.8	Setting Parameters in 7963_UPG_PARAMS.txt	A-6
A.2	Setting Source System-Specific Parameters in Informatica Parameter Files	A-6
A.2.1	Setting Parameters in 796_UPG_PARAMS.txt for Oracle EBS 11i Source Systems..	A-7
A.2.2	Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems	A-7
A.2.3	Setting Parameters in 7962_UPG_PARAMS.txt for PeopleSoft Source Systems	A-8
A.2.4	Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems	A-9
A.3	Setting Application-Specific Parameters in Informatica Parameter Files	A-9
A.3.1	Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics	A-10
A.3.1.1	Setting Parameters for Value Set Hierarchies and FSG Hierarchies	A-10
A.3.1.2	Setting Parameters for GL Data Extraction	A-10
A.3.1.2.1	Setting GL Data Extraction Parameters for Oracle EBS 11i Sources	A-10
A.3.1.2.2	Setting GL Data Extraction Parameters for Oracle EBS R12 Sources	A-11
A.3.1.3	Setting the COGS Fact Mapping for Oracle EBS R12	A-11
A.3.1.4	Setting the \$\$Hint1 Parameter for Oracle Databases	A-12
A.3.2	Setting Parameters in 796_UPG_PARAMS.txt for Oracle Project Analytics	A-12
A.3.3	Setting Parameters in 796_UPG_PARAMS.txt for Oracle Supply Chain and Order Management Analytics	A-13
A.3.3.1	Setting the TIME_GRAIN Parameter for Sales Order Lines Aggregate Fact and Invoice Lines Aggregate Fact Tables	A-13
A.3.3.2	Setting the \$\$PERIOD Parameter for the Customer Status History Fact Table	A-13
A.3.4	Setting Parameters and Mappings in 7962_UPG_PARAMS.txt for Oracle Financial Analytics	A-13
A.3.4.1	Setting GL Data Extraction Parameters for Oracle EBS 11i Sources	A-13
A.3.4.2	Setting GL Data Extraction Parameters for Oracle EBS 12 Sources	A-14
A.3.4.3	Setting GL Data Extraction Parameters for PeopleSoft Sources	A-14
A.3.4.4	Setting the COGS Fact Mapping for Oracle EBS R12	A-15
A.3.5	Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Financial Analytics	A-15
A.3.5.1	Setting Parameters for All Source Systems	A-16
A.3.5.2	Setting Parameters Specific to PeopleSoft	A-16
A.3.5.3	Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J	A-16
A.3.6	Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Human Resources Analytics	A-17

A.3.7	Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System	A-17
A.3.8	Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems	A-18

B Generating and Running Scripts to Drop and Create Indexes

B.1	Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2	B-1
B.2	Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2.....	B-2
B.3	Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2	B-3
B.4	Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3	B-3
B.5	Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3	B-4

C Creating Indexes for Oracle Financial Analytics

C.1	Indexes for SIL_PositionDimension_UPD7962 Mapping	C-1
C.2	Indexes for SIL_PositionDimensionHierarchy_UPD7962 Mapping	C-1

D Setting Database-Specific Parameters for the reset_infa_seq_gen.bat File

D.1	Running reset_infa_seq_gen.bat on Base Informatica Repository	D-1
D.1.1	Entering Values for the reset_infa_seq_gen.bat File on the Base Informatica Repository	D-1
D.1.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-4
D.1.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-5
D.2	Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle	D-6
D.2.1	Entering Values for the reset_infa_seq_gen_ORA.bat File on the Base Informatica Repository	D-6
D.2.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-8
D.2.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-9
D.3	Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server	D-10
D.3.1	Entering Values for the reset_infa_seq_gen_MSSQL.bat File on the Base Informatica Repository	D-10
D.3.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-11
D.3.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-12
D.4	Running reset_infa_seq_gen.bat on Upgrade Informatica Repository	D-13
D.4.1	Entering Values for the reset_infa_seq_gen.bat File on the Upgrade Informatica Repository	D-14
D.4.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-16
D.4.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-17
D.5	Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle	D-18
D.5.1	Entering Values for the reset_infa_seq_gen_ORA.bat File on the Upgrade Informatica Repository	D-18
D.5.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-19

D.5.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-21
D.6	Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server	D-22
D.6.1	Entering Values for the reset_infa_seq_gen_MSSQL.bat File on the Upgrade Informatica Repository	D-22
D.6.2	Obtaining ROW_WID from W_ORG_D and W_PRD_OF_WRK_BAND_D.....	D-23
D.6.3	Obtaining ROW_WID from W_PARTY_D and W_PRD_OF_WRK_BAND_D	D-24

E Using the DDLimp Utility

E.1	Using the DDLimp Utility to Run CTL Files	E-1
E.2	Additional DDLimp Commands.....	E-2

F Upgrading the Oracle BI Repository for Industry-Specific Analytics Applications

F.1	Common Dimensions.....	F-1
F.2	Merging Siebel Analytics and Oracle BI Repositories.....	F-2
F.2.1	Creating Working Folders.....	F-4
F.2.2	Trimming Repository Files.....	F-4
F.2.2.1	Trimming the Original Base Repository Files	F-5
F.2.2.2	Trimming the New Base Repository File	F-5
F.2.2.3	Trimming the Custom Repository Files	F-5
F.2.3	Renaming Objects in the Original Pharma Base Repository File.....	F-5
F.2.4	Equalizing the Oracle BI Repositories	F-6
F.2.5	Merging the Core Repository Files	F-7
F.2.6	Merging the Pharma Repository Files	F-9
F.2.7	Making Merge Decisions in the Administration Tool	F-10
F.3	Replacing Common Dimensions After the Repository Merge.....	F-10

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, JD Edwards, 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:

- Oracle Financial Analytics
- Oracle Human Resources Analytics
- Oracle Supply Chain and Order Management Analytics
- Oracle Procurement and Spend Analytics
- Oracle Project Analytics
- Oracle Sales Analytics
- Oracle Service Analytics
- Oracle Contact Center Telephony Analytics
- Oracle Marketing Analytics
- Oracle Loyalty Analytics
- Oracle Price Analytics
- Oracle Pharma Marketing Analytics
- Oracle Pharma Sales Analytics

For more details on the applications included in this release of Oracle BI Applications, see the *Oracle Business Intelligence Applications Licensing and Packaging Guide*. This guide is included in the Oracle Business Intelligence Media Pack. Also, see the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*, available at available at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/bi-apps-098545.html>.

The *Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users* is part of the documentation set for Oracle BI Applications. This guide contains

information about upgrading to Oracle BI Applications from different versions of Siebel Business Analytics and Oracle BI Applications.

Oracle recommends reading the *Oracle Business Intelligence Applications Release Notes* before installing, using, or upgrading Oracle BI Applications. The most current version of the *Oracle Business Intelligence Applications Release Notes* is available on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/bi-apps-098545.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

Audience

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

Documentation Accessibility

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

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Access to Oracle Support for Hearing-Impaired Customers

Oracle customers have access to electronic support through My Oracle Support or by calling Oracle Support at 1.800.223.1711. Hearing-impaired customers in the U.S. who wish to speak to an Oracle Support representative may use a telecommunications relay service (TRS). Information about the TRS is available at

<http://www.fcc.gov/cgb/consumerfacts/trs.html>, and a list of telephone numbers is available at <http://www.fcc.gov/cgb/dro/trsphonebk.html>.

International hearing-impaired customers should use the TRS at +1.605.224.1837. An Oracle Support engineer will respond to technical issues according to the standard service request process.

Related Documents

For more information, see the following Oracle BI Applications documentation set (available at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/bi-apps-098545.html>):

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

Also see the Oracle Business Intelligence Data Warehouse Administration Console documentation set (available at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>):

- *Oracle Business Intelligence Data Warehouse Administration Console Release Notes*
- *System Requirements and Supported Platforms for Oracle Business Intelligence Data Warehouse Administration Console*
- *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*
- *Oracle Business Intelligence Data Warehouse Administration Console Installation, Configuration and Upgrade Guide*

Conventions

The following text conventions are used in this document:

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

What's New in This Release

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

1.1 What's New in Oracle BI Applications Release 7.9.6.3

Oracle Business Intelligence Applications release 7.9.6.3 includes the following new features:

Certification With Oracle Business Intelligence Enterprise Edition 11g (11.1.1.5.0)

Oracle Business Intelligence Applications release 7.9.6.3 is certified with Oracle Business Intelligence Enterprise Edition release 11g (11.1.1.5.0). If you are using an earlier version of Oracle BI Enterprise Edition, you must upgrade to release 11.1.1.5.0 in order to use Oracle BI Applications release 7.9.6.3.

Oracle Business Intelligence Enterprise Edition release 11g (11.1.1.5.0) includes significant enhancements, including a change in security strategy, and many behavioral changes and enhancements to dashboards and reports. You should familiarize yourself with the changes by reviewing the "New Features" sections in the guides in the Oracle Business Intelligence Enterprise Edition 11g release 1 (11.1.1.5.0) documentation set. In addition, for a summary of the changes in behavior and appearance, see the section titled "Appendix B: Possible Changes in Oracle BI Enterprise Edition Appearance and Behavior After Upgrade," in *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)*.

Before performing the platform upgrade, you should thoroughly review *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)*, and carefully read the chapter titled, "Planning to Upgrade from Oracle BI 10g to BI 11g."

Improved Support for Teradata

Improved upgrade support for customers whose Oracle Business Analytics Warehouse is on a Teradata database and Informatica Repository on either an Oracle or SQL Server database.

Improved support for Oracle EBS

Improved support for Oracle EBS customers upgrading the Oracle Business Analytics Warehouse because of transactional database upgrades from version 11i to R12.x.

Support for Informatica Version 9.0.1 Hotfix 2

Oracle BI Applications requires installation of Informatica version 9.0.1 Hotfix 2.

Overview of Oracle BI Applications Upgrade

This chapter provides an overview of the Oracle Business Intelligence Applications upgrade process. It includes the following topics:

- [Section 2.1, "Supported Upgrade Paths"](#)
- [Section 2.2, "Why Upgrade to Oracle BI Applications Release 7.9.6.3?"](#)
- [Section 2.3, "Considerations About Upgrading to Oracle BI Enterprise Edition 11g"](#)
- [Section 2.4, "Considerations About Upgrading Oracle BI Applications"](#)
- [Section 2.5, "Best Practices for Preparing to Upgrade"](#)
- [Section 2.6, "Major Stages of Oracle BI Applications Upgrade"](#)

Note: Upgrading Oracle BI Applications requires careful planning and testing. You must allocate sufficient time for implementing the upgrade. It is highly recommended that customers consider obtaining support from Oracle Consulting Services to assist with an applications upgrade.

Also note that the process of upgrading to Oracle BI Applications release 7.9.6.3 includes an upgrade of the Oracle Business Intelligence platform to Oracle Business Intelligence Enterprise Edition 11g (11.1.1.5.0).

2.1 Supported Upgrade Paths

The *Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users* provides instructions for upgrading to the current release of Oracle BI Applications from the previous releases listed below:

- **Siebel Analytics 7.5.3**
For instructions, see [Chapter 3, "Upgrading Siebel Analytics 7.5.x."](#)
- **Siebel Business Analytics 7.7, 7.7.x and 7.8.x**
For instructions, see [Chapter 4, "Upgrading Siebel Business Analytics 7.7, 7.7.x and 7.8.x."](#)
- **Siebel Business Analytics for Life Sciences 7.8.0, 7.8.1 and 7.8.2**
For instructions, see [Chapter 4, "Upgrading Siebel Business Analytics 7.7, 7.7.x and 7.8.x."](#)
- **Oracle Business Intelligence Applications 7.9.0 through 7.9.6.2**

For instructions, see one of the following chapters, depending on your source system:

- [Chapter 5, "Upgrading Oracle BI Applications 7.9.x for Siebel Source Systems"](#)
- [Chapter 6, "Upgrading Oracle BI Applications for Oracle Source Systems"](#)
- [Chapter 7, "Upgrading Oracle BI Applications for PeopleSoft Source Systems"](#)
- [Chapter 8, "Upgrading Oracle BI Applications for JD Edwards Source Systems"](#)

2.2 Why Upgrade to Oracle BI Applications Release 7.9.6.3?

There can be many reasons your organization would consider upgrading to Oracle BI Applications release 7.9.6.3, including, but not limited to, the following:

- An upgrade of the transactional applications can trigger a need to upgrade Oracle BI Applications to the latest release.
 - However, it is not always necessary to upgrade Oracle BI Applications if you upgrade the transactional applications.
 - For example, if both the version of transactional applications from which you are upgrading and to which you are upgrading are supported by the release of Oracle BI Applications you are using, you do not need to upgrade Oracle BI Applications to the latest release
- Strong need to leverage new features and enhancements offered by Oracle BI Enterprise Edition 11g (11.1.1.5.0), including support for newer versions of databases.
- New features and enhancements offered by Oracle BI Applications, including new application content, support for newer versions of source systems, support for new versions of databases, and language support.
- Potential cost savings on additional maintenance cost in the event that support is no longer available for the transactional applications, the database version, or the Informatica version.
- Business requirements.

2.2.1 Upgrade Options

The most common triggers for upgrading Oracle BI Applications are an upgrade of the underlying transactional source system or systems, the need to take advantage of new content in the latest version of Oracle BI Applications, and support for new database platforms or operating systems.

While upgrading only the Oracle Business Intelligence Foundation to leverage several enhancements offered in Oracle BI Enterprise Edition 11g (11.1.1.5.0) is a supported option, note that only Oracle BI Applications release 7.9.6.3 is certified on Oracle BI Enterprise Edition 11g (11.1.1.5.0). Previous releases of Oracle BI Applications are not certified on Oracle BI Enterprise Edition 11g. Unless you have a compelling need to leverage the enhancements offered in Oracle BI Enterprise Edition 11g (11.1.1.5.0), suitable options worth considering are 1) staying on Oracle BI Enterprise Edition 10g, or 2) upgrading Oracle BI Enterprise Edition as part of an integral upgrade along with Oracle BI Applications.

As with any upgrade, Oracle recommends that Oracle BI Applications customers who currently use Oracle BI Enterprise Edition 10g do a thorough cost-benefit analysis before considering an upgrade to Oracle BI Enterprise Edition 11g (11.1.1.5.0).

2.3 Considerations About Upgrading to Oracle BI Enterprise Edition 11g

Upgrading to Oracle BI Applications release 7.9.6.3 requires an upgrade of the Oracle Business Intelligence Foundation to Oracle BI Enterprise Edition 11g (11.1.1.5.0).

Upgrading Oracle Business Intelligence from 10g to 11g is a substantial project, requiring the need to budget sufficient time and resources for analysis, implementation, and verification of the upgrade. Oracle provides tools and technology to automate much of the upgrade process. However, the precise strategy that you will want to adopt will depend both on the configuration of the existing 10g system, and on the required configuration of the upgraded 11g system. Before performing the platform upgrade, you should thoroughly review *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)*, and carefully read the chapter titled, "Planning to Upgrade from Oracle BI 10g to BI 11g."

Oracle BI Applications customers upgrading to the Oracle Business Intelligence Foundation to Oracle BI Enterprise Edition 11g (11.1.1.5.0) should note that the BI Server in Oracle BI Enterprise Edition 11g (11.1.1.5.0) is more stringent in its metadata validation checks and produces a number of warnings on the repositories that are upgraded from earlier Oracle BI Applications releases. These warnings can be fixed manually. For more information about these warning and for the workarounds to fix the issues, see *Oracle Business Intelligence Applications Release Notes*.

In addition, Oracle BI Enterprise Edition 11g (11.1.1.5.0) includes changes and enhancements that affect the look, feel, and behavior of dashboards and reports. For a description of these changes, see the section titled "Appendix B: Possible Changes in Oracle BI Enterprise Edition Appearance and Behavior After Upgrade," in *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)*.

You should also be aware of the following security-related changes in Oracle BI Enterprise Edition 11g (11.1.1.5.0):

- Repositories now have repository-specific passwords that are used to encrypt the repository contents. The repository password is stored in an external credential store when you publish a repository in Fusion Middleware Control, so that the Oracle BI Server can retrieve the password to load the repository.
- Groups no longer exist in the repository as objects. Instead, you implement data access security based on the application roles to which a user belongs.
- Application roles are managed in an external policy store. Application role objects exist in the repository, but these objects are pointers (references) to the externally managed roles.
- Users are managed in an external authentication provider and are no longer managed in the repository. User objects exist in the repository, but these objects are pointers (references) to the externally managed users.

See *Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition* for more information about these and other security-related changes.

2.4 Considerations About Upgrading Oracle BI Applications

You should consider the following points before you begin your upgrade:

- Review the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* to make sure your database platform versions are supported.
- The upgrade process is not only a technical process. It requires significant planning and involvement from many teams, including development, database administrators, business analysts, QA, and so on.
- You should determine what customizations were made to your existing application before you begin the upgrade process. It is recommended that you budget sufficient time (in the upgrade project) for detailed review of current customizations and relevance of the customizations for the new version of the application to which you plan to upgrade.
- You will also need to analyze the impact of the schema changes on your current custom implementation. The extent of your customizations will have an impact on the length of time required for the upgrade.
- At different stages of the upgrade you will also need to analyze your organization's business requirements. For example, at the stage in which you are upgrading the Oracle Business Intelligence repository, you will need to know which repository objects are not in use so that they can be trimmed from the repository. To determine this information, you will need to know what reports are in use.
- Depending on the release from which you are upgrading, moving the customizations in your existing repositories, reports, and dashboards to the new version may require a multi-step process and may involve manual processes at some stages.
- It is highly recommended that you use side-by-side environments when performing each stage of the upgrade process. Enabling side-by-side instances of the entire Oracle BI Applications environment is a critical success factor for upgrade.

For some stages of the upgrade, you can upgrade your environments in place. However, for comparison and benchmarking purposes, it is recommended that you upgrade using side-by-side environments.
- For each stage of the upgrade process, you need to allocate a reasonable amount of time to validate the results of that stage and address any problems. In addition, final user acceptance testing must confirm that the entire upgrade process was successful.
- When you move from a development environment to test or production environments, you must perform the following stages of the upgrade process for each environment:
 - Upgrade the Oracle Business Intelligence platform
 - Upgrade the Oracle BI Applications
 - Migrate the data
- You can migrate the DAC, Informatica, and Oracle BI repositories, and the presentation catalogs into the production environment after you merge and test them.

2.5 Best Practices for Preparing to Upgrade

This section provides best practice steps you need to perform before you begin the upgrade process.

- Review this guide, *Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users*, in its entirety.
- Rename and copy into a different location your current repositories, presentation catalog (formerly known as the Web Catalog), and DAC installation folder from your existing environment. The file names and locations listed below represent the standard names and locations of files that shipped with previous versions of Siebel Business Analytics Applications and Oracle BI Applications.

When you back up the DAC Repository, you export the DAC metadata, in XML format (using the DAC's Export tool), into a different database, which can be used later as a reference. For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

Type	File or Folder	Default BI 10g Location
DAC metadata repository	*.xml	OracleBI\dac\export Note: This is the directory into which you copied the DAC metadata files after you installed DAC.
Oracle BI Repository (formerly called Siebel Analytics Repository)	Oracle BI Repository (or Siebel Analytics Repository) you are currently using	OracleBI\server\Repository
Informatica Repository	Informatica Repository file (for example, Oracle_BI_DW_Base.rep)	OracleBI\dwrep\Informatica\Repository
Presentation Catalog (formerly called Web Catalog)	Presentation Catalog (or Web Catalog) you are currently using	OracleBIData\web\catalog
DAC installation folder	dac	OracleBI\dac

- Gather information that describes the current implementation, including the following:

- Version of the transactional database to which you upgraded.
- Version of the Oracle BI Enterprise Edition (platform) or Siebel Business Analytics platform from which you are upgrading.

To determine the platform version from which you are upgrading, launch the Administration Tool, and click Help, and then click Administration Tool in the toolbar. A message box provides the platform version.

- Version of Siebel Business Analytics Applications or Oracle BI Applications from which you are upgrading.

To determine the version of Siebel Business Analytics Applications or Oracle BI Applications from which you are upgrading, launch the Data Warehouse Administration Console (DAC), and click Help, and then click About DAC.

Note: If DAC is not a component of your Analytics installation, you are running Siebel Analytics 7.5.x.

2.6 Major Stages of Oracle BI Applications Upgrade

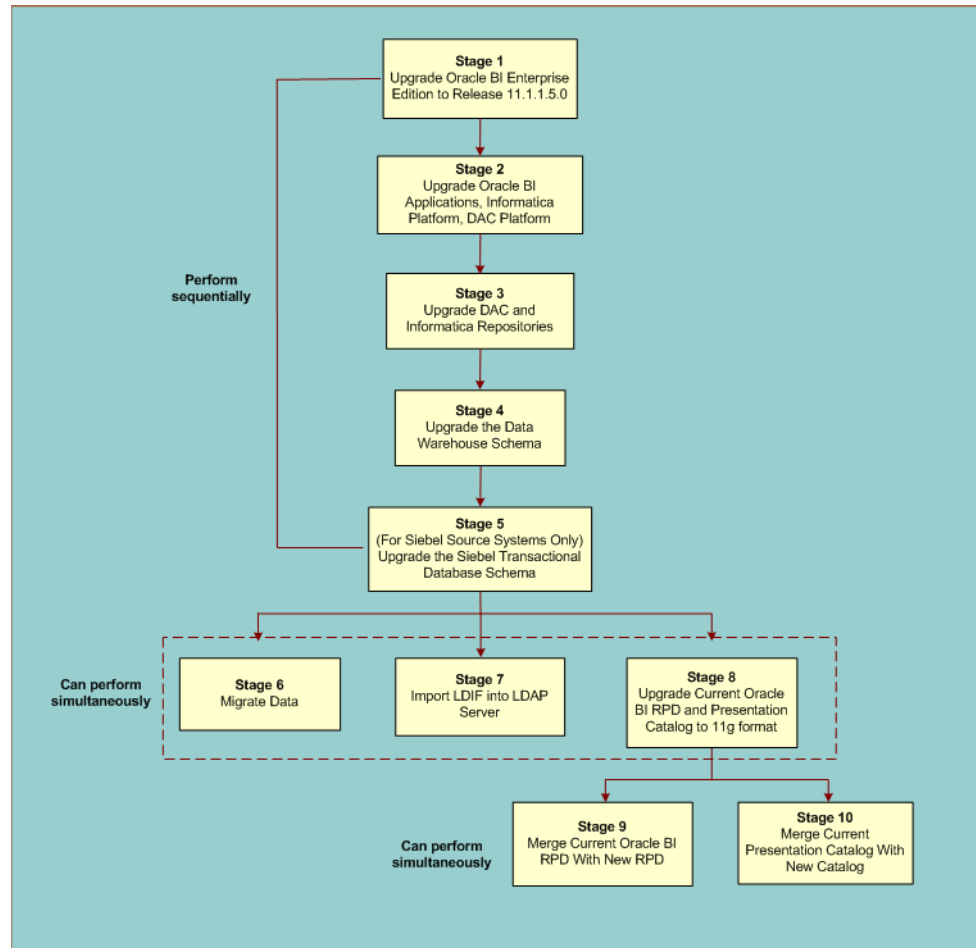
This section describes the major stages of the Oracle BI Applications upgrade process and includes the following topics:

- [Section 2.6.1, "Sequence of Upgrade Stages"](#)
- [Section 2.6.2, "Documentation Related to the Major Stages of the Upgrade Process"](#)

Note: iBots (agents) are not included in the standard (out-of-the-box) Oracle BI Applications product. However, you can upgrade a 10g Scheduler schema to 11g. For more information, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)*.

2.6.1 Sequence of Upgrade Stages

[Figure 2–1](#) shows the upgrade flow for the major stages of the Siebel Business Analytics and Oracle BI Applications upgrade process. Depending on the prior release from which you are upgrading, you may find minor changes in the process shown in [Figure 2–1](#).

Figure 2–1 Oracle BI Applications Upgrade Flow

In general, you must complete stages one through five sequentially, in the following order:

- **Stage 1:** Upgrade the Siebel Business Analytics platform or Oracle BI platform to Oracle Business Intelligence Enterprise Edition Release 11g (11.1.1.5.0). See [Section 2.3, "Considerations About Upgrading to Oracle BI Enterprise Edition 11g"](#) for important information about this upgrade process.
- **Stage 2:** Upgrade the following:
 - Siebel Business Analytics Applications or Oracle BI Applications to the current release of Oracle BI Applications
 - Informatica PowerCenter to version 9.01 Hotfix 2
 - DAC platform to release 10.1.3.4.1
- **Stage 3:** Upgrade the metadata in the DAC and Informatica repositories.
- **Stage 4:** Upgrade the data warehouse schema.
- **Stage 5:** (For Siebel source systems) upgrade the Siebel transactional database schema.

After completing stage 4 or stage 5 for Siebel source systems, you can then perform the next three stages simultaneously:

- **Stage 6:** Migrate data into the upgraded data warehouse. Note that for testing purposes, you can migrate a subset of data and study the results before migrating all of the data.
- **Stage 7:** Import the Oracle BI Applications release 7.9.6.3 LDIF file into the embedded LDAP Server.
- **Stage 8:** Use the Oracle BI Enterprise Edition 11g Upgrade Assistant to upgrade the following repositories and presentation catalog (or Siebel Analytics Web catalog) from the 10g format to the 11g format:
 - The Oracle BI repository you are currently using (the repository that includes your customizations).
 - The standard (out-of-the-box) Oracle BI repository version 7.9.x (the version from which you are upgrading).
 - The presentation catalog you are currently using.

Note: Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

After you complete stage 8, you can perform stages 9 and 10 simultaneously:

- **Stage 9:** Merge your customizations of a prior release of the Oracle BI repository with the new release of the Oracle BI repository.

See the section titled "Understanding the Upgrade of Repository Metadata," in the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)* for information about Oracle BI Enterprise Edition platform behavior related to the upgrade of the Oracle BI repository.

- **Stage 10:** Merge your customizations of a prior release of the presentation catalog with the new release of the presentation catalog.

See the section titled "Understanding Oracle BI Presentation Catalog Upgrade," in the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence 11g Release 1 (11.1.1)* for information about Oracle BI Enterprise Edition platform behavior related to the upgrade of the Siebel Analytics Web Catalog and the Oracle BI Presentation Catalog.

Note: Depending on the prior release from which you are upgrading, you may find minor changes in the process described in this section.

Note: The schema and data are upgraded in place and affect only the preconfigured (out of the box) content. Customizations to the schema are not affected by the upgrade process.

2.6.2 Documentation Related to the Major Stages of the Upgrade Process

Table 2–1 provides the documentation you should refer to for each stage of the upgrade.

Table 2–1 Major Stages of the Oracle BI Applications Upgrade and Related Documentation

Stage	Documentation Related to Stage
Stage 1: Upgrade the Siebel Business Analytics platform or Oracle BI Enterprise Edition platform to Oracle BI Enterprise Edition Release 11g (11.1.1.5.0).	<ul style="list-style-type: none"> ■ <i>Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence</i> <p>Note: You should carefully review the chapter titled, "Planning to Upgrade from Oracle BI 10g to BI 11g" before beginning the upgrade process.</p>
Stage 2: Upgrade the following: <ul style="list-style-type: none"> ■ Siebel Business Analytics Applications or Oracle BI Applications to the current release of the Oracle BI Applications ■ Informatica PowerCenter ■ DAC platform 	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i> <p>This guide (the guide you are currently reading) provides instructions for the upgrade based on the source system. Follow the steps described in the appropriate chapter for your source system:</p>
Stage 3: Upgrade the metadata in the DAC and Informatica repositories.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i> ■ <i>Oracle Business Intelligence Data Warehouse Administration Console User's Guide</i>
Stage 4: Upgrade the data warehouse schema.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i>
Stage 5: (For Siebel source systems only) Upgrade the Siebel transactional database schema.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i>
Stage 6: Migrate data into the upgraded data warehouse.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i>
Stage 7: Import the Oracle BI Applications release 7.9.6.3 LDIF file into the embedded LDAP Server.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i>
Stage 8: Upgrade the Oracle BI repositories and presentation catalog from the 10g format to the 11g format.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i>
Stage 9: Merge the Siebel Analytics repository or Oracle BI repository with the new version installed with the current release.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i> ■ <i>Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence</i>
Stage 10: Merge the Siebel Business Analytics Web Catalog or Oracle BI presentation catalog with the new version installed with the current release.	<ul style="list-style-type: none"> ■ <i>Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users</i> ■ <i>Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence</i>

Part I

Upgrading When Your Source System is Siebel CRM

Part I contains instructions for upgrading to the current release of Oracle BI Applications from previous releases of Oracle BI Applications, Siebel Business Analytics, and Siebel Analytics.

Part I contains the following sections:

- [Chapter 3, "Upgrading Siebel Analytics 7.5.x"](#)
Follow the instructions in this section for upgrading to the current release of Oracle BI Applications from Siebel Analytics 7.5.x
- [Chapter 4, "Upgrading Siebel Business Analytics 7.7, 7.7.x and 7.8.x"](#)
Follow the instructions in this section for upgrading to the current release of Oracle BI Applications from Siebel Business Analytics 7.7, 7.7.x, and 7.8.x.
- [Chapter 5, "Upgrading Oracle BI Applications 7.9.x for Siebel Source Systems"](#)
Follow the instructions in this section for upgrading to the current release from previous releases of Oracle BI Applications if your source system is Siebel CRM.

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

Upgrading Siebel Analytics 7.5.x

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

This section contains instructions for upgrading Oracle's Siebel Analytics release 7.5.x.

This section includes the following topics:

- [Section 3.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 3.2, "Upgrading Oracle BI Applications"](#)
- [Section 3.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 3.4, "Upgrading the Informatica Repository"](#)
- [Section 3.5, "Configuring Informatica PowerCenter Version 9.0.1 to Work With Oracle BI Applications and DAC"](#)
- [Section 3.6, "Installing and Configuring DAC"](#)
- [Section 3.7, "Overview of Upgrading the Data Warehouse Schema and Migrating Data"](#)
- [Section 3.8, "Upgrading the Data Warehouse Schema to Version 7.7 and Migrating Data"](#)
- [Section 3.9, "Upgrading the Data Warehouse Schema to Version 7.9.0"](#)
- [Section 3.10, "Importing New Schema Definitions into the Siebel Transactional Database"](#)
- [Section 3.11, "Migrating Data into the Data Warehouse Upgraded to Version 7.9.0"](#)
- [Section 3.12, "Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data"](#)

- Section 3.13, "Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data"
- Section 3.14, "Upgrading the Data Warehouse Schema to Version 7.9.5.1"
- Section 3.15, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data"
- Section 3.16, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1"
- Section 3.17, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data"
- Section 3.18, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data"
- Section 3.19, "Localizing the Policy Store"
- Section 3.20, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"
- Section 3.21, "Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version"
- Section 3.22, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"
- Section 3.23, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog"
- Section 3.24, "Regression Testing the Oracle BI Repository Merge"

3.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

3.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release.)

This will be the backup of your DAC Client and Server and the DAC metadata files.)

3.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ significantly from previous versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 3.4, "Upgrading the Informatica Repository."](#)

In [Section 3.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle_BI_DW_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle_BI_DW_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

3.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

Note: This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 3.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle_BI_DW_Base repository that you restore in the steps below.

2. Copy the Oracle_BI_DW_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.

Note: The Oracle_BI_DW_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 3.2, "Upgrading Oracle BI Applications."](#)

3. Restore the Oracle_BI_DW_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
 - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
 - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.
 - c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

3.5 Configuring Informatica PowerCenter Version 9.0.1 to Work With Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

3.6 Installing and Configuring DAC

Siebel Analytics release 7.5.x does not include the Data Warehouse Administration Console (DAC). In the current release of Oracle BI Applications, you will use the DAC Client and Server and the DAC Repository. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

To install and configure DAC, follow the instructions in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. Make sure you perform the tasks in the following sections of the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Preinstallation Tasks
- Installing and Setting Up the DAC Platform
- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

Note: After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring Relational Connections in Informatica Workflow Manager
- Configuring the SiebelUnicodeDB Custom Property
- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

3.7 Overview of Upgrading the Data Warehouse Schema and Migrating Data

The process for upgrading the data warehouse schema and migrating data for Siebel Business Analytics releases 7.5.x involves multiple phases, as described below.

Note: The Oracle BI Applications upgrade logic is coded such that you must complete each of the phases listed below. To upgrade to the current release of Oracle BI Applications, you cannot skip any of the phases.

- **Phase 1**

In phase 1, you will upgrade the data warehouse schema to version 7.7 and migrate data into the upgraded data warehouse. For instructions, see [Section 3.8](#).

- **Phase 2**

In phase 2, you will perform the following steps:

1. Upgrade the data warehouse schema to version 7.9.0. For instructions, see [Section 3.9](#).

2. Import new schema definitions into the Siebel transactional database. For instructions, see [Section 3.10](#).
 3. Migrate data into the data warehouse upgraded to version 7.9.0. For instructions, see [Section 3.11](#).
- **Phase 3**
In phase 3, you will upgrade your data warehouse schema to version 7.9.4 and migrate data into the upgraded data warehouse. For instructions, see [Section 3.12](#).
 - **Phase 4**
In phase 4, you will upgrade your data warehouse schema to version 7.9.5 and migrate data into the upgraded data warehouse. For instructions, see [Section 3.13](#).
 - **Phase 5**
In phase 5, you will upgrade your data warehouse schema to version 7.9.5.1. For instructions, see [Section 3.14](#).
 - **Phase 6**
In phase 6, you will do one of the following:
 - Upgrade your data warehouse schema from version 7.9.5.1 to version 7.9.6.1 and migrate data. For instructions, see [Section 3.15](#).
 - If you are currently running Oracle BI Applications version 7.9.6 and using an Oracle database, you can upgrade your data warehouse schema to version 7.9.6.1 and migrate data. For instructions, see [Section 3.16](#).
 - **Phase 7**
In phase 7, you will upgrade your data warehouse schema to version 7.9.6.2 and migrate data into the upgraded data warehouse. For instructions, see [Section 3.17](#).
 - **Phase 8**
In phase 8, you will upgrade your data warehouse schema to version 7.9.6.3 and migrate data into the upgraded data warehouse. For instructions, see [Section 3.18](#).

3.8 Upgrading the Data Warehouse Schema to Version 7.7 and Migrating Data

Follow this procedure to upgrade the data warehouse schema to version 7.7 and migrate data.

To upgrade the data warehouse schema

1. Run the schema upgrade script.
This script adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\CTLFiles, and locate the ddl sme_HOR_77.ctl file (Horizontal) or ddl sme_SIA_77.ctl file (Vertical).
 - b. If you changed any of the preconfigured column definitions in the CTL file for your implementation, you need to edit the CTL file to reflect the change. For example, if you renamed or resized an existing column in W_DAY_D, you need to make the same change in the CTL file. If you do not make the change in the CTL file, when the CTL file is run, the column definition will revert to the preconfigured definition. (As a customization, if a column size was

changed from VARCHAR(50) to VARCHAR(100), and the same change was not made in the CTL file before running, the column will revert to the preconfigured value of VARCHAR(50), which could cause data to be truncated in some databases.)

- c. Use the DDLimp utility to run one of the following scripts:

For Siebel Applications (Horizontal), run `ddlsme_HOR_77.ctl`.

For Siebel Industry Applications (Vertical), run `ddlsme_SIA_77.ctl`.

Use the following command, substituting the correct script name where appropriate.

```
..\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC connect string>
/G SSE_ROLE /I N /R Y /F <..\OracleBI\dwrep\Upgrade\CTLFiles\ddlsme_HOR_77.ctl>
/L <..\oracleBI\dwrep\ddlsme_HOR_77.log>
```

For example:

```
DDLIMP /U SADMIN /P SADMIN /C SIEBEL_OLTP /G SSE_ROLE
/I N /R Y /F C:\OracleBI\dwrep\Upgrade\CTLFiles\ddlsme_HOR_77.ctl /L
C:\OracleBI\dwrep\ddlsme_HOR_77.log
```

Notes:

- * /P <PASSWORD> - The password for the data warehouse.
 - * /C <ODBC connect string> - The name of the ODBC connect string.
 - * /I N - Tells DDLimp to ignore the indexes DDL defined in the CTL file if any exist. It does not change existing indexes. (The default is /I Y, which tells DDLimp to create and merge indexes from the CTL file with the indexes in the database.
 - * For Oracle databases, use the Data Direct drivers.
 - * In addition, you can use the following commands:
 - * /W Y - If the OLAP database is Oracle and Unicode.
 - * /Z Y - If the OLAP database is DB2 or SQL Server and Unicode.
 - * /B <TABLE_SPACE_NAME> - If you want to create these tables in a separate table space. For DB2, This must be specified as 32K tablespace.
 - * /X <INDEX_TABLE_SPACE_NAME> - If you want to create the indexes in a separate tablespace. For DB2, This must be specified as 32K tablespace.
 - * /Y - Storage File for DB2/390.
 - * /R - Regrant tables.
2. Copy the file `Upgrade.rep` file from the folder `OracleBI\dwrep\Upgrade\Informatica\Repository` into the folder `<INFA_HOME>\server\infa_shared\Backup`.
 3. Launch the Informatica Administrator and restore `Upgrade.rep` (located in `<INFA_HOME>\server\infa_shared\Backup`) into a database other than the database in which you restored `Oracle_BI_DW_Base.rep`.
 4. Navigate to the folder `OracleBI\dwrep\Upgrade\Informatica\ParameterFiles` and copy the file `753_TENERIFE_UPG_PARAMS.txt` into the `SrcFiles` folder on the

Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.

5. Rename 753_TENERIFE_UPG_PARAMS.txt to TENERIFE_UPG_PARAMS.txt.
6. In the Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and edit the connect string, user and password for the relational connections as follows:
 - a. Edit the PARAM_OLTP_SIEBEL connection to point to the Siebel transactional database.
 - b. Edit the DataWarehouse connection to point to the newly upgraded data warehouse database.
 - c. Create or edit the PARAM_DAC_OLD connection to point to the previous DAC Repository database (the version from which you are upgrading).
 - d. Create or edit the PARAM_DAC_NEW connection to point to the DAC Repository database.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

7. For Siebel Applications (Horizontal), in Informatica Workflow Manager, navigate to the folder UPGRADE_753_to_770_HOR, and execute the following workflows in the order indicated:
 - a. Upgrade_R_Image
 - b. Upgrade_Unspecifieds
 - c. Upgrade_Agree
 - d. Upgrade_Asset
 - e. Upgrade_Opty
 - f. Upgrade_Order
 - g. Upgrade_Quote
 - h. Upgrade_Response
 - i. Upgrade_ServiceRequest
 - j. Upgrade_Others
 - k. Upgrade_Visibility
 - l. UpgradeSlowlyChangingDimensionStartDates
 - m. DAC_Metadata_Upgrade_Workflow
8. For Siebel Industry Applications (Vertical), in Informatica Workflow Manager, navigate to the folder UPGRADE_753_to_770_SIA, and execute the following workflows in the order indicated:
 - a. Upgrade_R_Image
 - b. Upgrade_Unspecifieds
 - c. Upgrade_Agree

- d. Upgrade_Asset
- e. Upgrade_Opty
- f. Upgrade_Order
- g. Upgrade_Quote
- h. Upgrade_Response
- i. Upgrade_ServiceRequest
- j. Upgrade_Others
- k. Upgrade_Visibility
- l. UpgradeSlowlyChangingDimensionStartDates
- m. Upgrade_Industry_R_Image
- n. Upgrade_Industry_Unspecified
- o. Upgrade_LS_ActivityProduct
- p. Upgrade_LS_Others
- q. Upgrade_Industry_Household
- r. Upgrade_FINS_Visibility
- s. UpgradeSlowlyChangingDimensionStartDates_Industry
- t. DAC_Metadata_Upgrade_Workflow

3.9 Upgrading the Data Warehouse Schema to Version 7.9.0

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

To upgrade the data warehouse schema

1. Run the UPGRADE.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the 790_UPGRADE_PRE_CTL_SCRIPT.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
 - c. Open the 790_UPGRADE_PRE_CTL_SCRIPT.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
3. Run the DW.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the 790_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.

3.10 Importing New Schema Definitions into the Siebel Transactional Database

This procedure upgrades transactional database objects that relate to Oracle Business Intelligence Applications. It does not upgrade transactional database objects for Siebel CRM applications.

To import new schema definitions into the Siebel transactional database

1. Import schema definitions for non-image tables:
 - a. Access the DDLimp utility.
 - b. Run the DDL_OLTP.ctf script that is located in the OracleBI\dwrep\Upgrade\CTLFiles folder.

Note: If you receive an error message stating a particular object already exists in the database, use the DDLimp Merge argument (/M Y) in the DDLimp command to resolve the error.
2. Import schema definitions for image tables:
 - a. In DAC, go to the Design view, and select your custom container from the drop-down list to the right of the Execute button.
 - b. Click the Tables tab.
 - c. Query for all tables for which the image suffix is not null.
 - d. Right-click in the list of tables returned by the query, and select Change Capture Scripts, and then Generate Image and Trigger Scripts.
 - e. In the Triggers and Image Tables dialog box, do the following:
 - Select the option All Tables in the List.
 - Select the option Generate Image Table Scripts.
 - Select the appropriate database type for the source system.
 - Click OK.
 - f. Open the SQL client for the source system database type, for example, SQL Plus for Oracle, Query Analyzer for SQL Server, or a command window for DB2.

The script may contain many lines; therefore, you can save the script file as a SQL file and execute it in a SQL client.
 - g. Copy the scripts generated by DAC into the SQL client and execute them.

3.10.1 Verifying the Siebel Transactional Database Upgrade

Follow this procedure to verify the following tables were created in the Siebel transactional database.

To verify the transactional database upgrade

- For all upgrade paths, verify the following tables were created in the Siebel transactional database:
 - S_ETL_R_IMG_XXX
 - S_ETL_I_IMG_XXX
 - S_ETL_D_IMG_XXX

- S_ETL_PARAM
- S_ETL_PRD_ATTR
- S_ETL_PRD_REL

3.11 Migrating Data into the Data Warehouse Upgraded to Version 7.9.0

Follow this procedure to migrate data into the data warehouse upgraded to version 7.9.0.

To migrate data into the upgraded data warehouse

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\SrcFiles and copy the *.csv files into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
2. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 770_TENERIFE_UPG_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
3. Rename 770_TENERIFE_UPG_PARAMS.txt to TENERIFE_UPG_PARAMS.txt.
4. In the file TENERIFE_UPG_PARAMS.txt:
 - a. Search for the parameter \$\$SourceConnection.
 - b. Set the value to one of the following options, based on your Siebel applications (OLTP) version:
 - * SEBL_63
 - * SEBL_753
 - * SEBL_771
 - * SEBL_78
 - * SEBL_80 (use this value for OLTP versions 8.x and 8.1.1)
 - * SEBL_VERT_753
 - * SEBL_VERT_771
 - * SEBL_VERT_78
 - * SEBL_VERT_80 (use this value for OLTP versions 8.x and 8.1.1)

For example, if your Siebel applications (OLTP) version is Siebel Industry Applications 7.5.x, the parameter should appear as:

```
$$SourceConnection=SEBL_VERT_753
```
5. In the file TENERIFE_UPG_PARAMS.txt:
 - a. Search for the parameter \$\$Source_Container.
 - b. Set the value to one of the following options, based on your Siebel applications (OLTP) version:
 - * Siebel 6.3
 - * Siebel 7.5.3
 - * Siebel 7.5.3 Vertical
 - * Siebel 7.7.1

- * Siebel 7.7.1 Vertical
- * Siebel 7.8
- * Siebel 7.8 Vertical
- * Siebel 8.0 (use this value for OLTP versions 8.x and 8.1.1)
- * Siebel 8.0 Vertical (use this value for OLTP versions 8.x and 8.1.1)

For example, if your Siebel applications (OLTP) version is Siebel Industry Applications 7.5.x, the parameter should appear as:

```
$$Source_Container=Siebel 7.5.3 Vertical
```

6. In the file TENERIFE_UPG_PARAMS.txt, edit the ETL_PROC_WID parameter as follows:

```
MPLT_GET_ETL_PROC_WID.$$ETL_PROC_WID=<latest ETL_PROC_WID value from your database>
```

You can get this value from W_PARAM_G.ETL_PROC_WID.

7. If you are running Siebel Industry Applications (Vertical), in the file TENERIFE_UPG_PARAMS.txt, set the VERTICAL_UPGRADE parameter to 1. For example:

```
$$VERTICAL_UPGRADE=1
```

8. For Siebel Industry Applications (Vertical), define the alignment rule to be used for ETL loads.

- a. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\SrcFiles and copy the file AlignmentType_LS_782.csv into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
- b. Rename the file AlignmentType_LS_782.csv to AlignmentType.csv.
- c. Open the file AlignmentType.csv, and enter one of the following alignment item types under the ALIGN_TYPE field:
 - * Zipcode
 - * Account/Contact
 - * Brick
- d. Save the file.

The Oracle Business Analytics Warehouse allows only one alignment type to be used for ETL loads during upgrade.

9. In Informatica Workflow Manager, navigate to the folder UPGRADE_770_to_79, and execute the following workflows in the order indicated:

- a. MARKETING_LOAD
- b. Upgrade_Dimensions_Industry
- c. Update_Dimensions
- d. Update_Dimension_Unspecified
- e. Update_Facts
- f. (For Siebel Industry Applications only) Upgrade_LS_Dimensions
- g. (For Siebel Industry Applications only) Upgrade_LS_Facts

- h. SIL_PositionDimensionHierarchy_Full
- i. Load_INT_ORG_DH
- j. DIMENSION_LOAD

Note: If you are using the SCD version of the dimension, replace the corresponding TENN_UPG_W_XXX_D_784_To_W_XXX_D session with TENN_UPG_W_XXX_D_784_SCD_To_W_XXX_D. This will upgrade the data from the W_XXX_SCD version of the dimension to the new SCD-enabled W_XXX_D dimension.

- k. DIMENSION_UNSPECIFIED_UPDATE
- l. FACT_UPDATE
- m. DAC_Metadata_Upgrade_Workflow

Note: You need to run this workflow for upgrading to the data warehouse schema version 7.9.0 even though you ran a workflow with the same name during the upgrade to version 7.7.

10. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, navigate to the folder UPGRADE_790_to_791_SBL80UPG and run the following workflows in the order indicated:
 - a. UPGRADE_DIMENSIONS
 - b. UPGRADE_FACTS
11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
12. To verify the data migrated successfully:
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the directory \Informatica PowerCenter 9.0.1\Server\SessLogs indicates errors or failures.
 - c. Check whether the SQL scripts (790_UPGRADE_PRE_CTL_SCRIPT.sql and 790_UPGRADE_PRE_DIMENSION_SCRIPT.sql) that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log files for the CTL files (Upgrade.ctl and DW.ctl) that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables.

13. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_784 tables, LKP_xxxx_x, and W_ASSET_D_TMP.

This step frees the space occupied by these backup tables.

3.12 Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data

Follow this procedure to upgrade the data warehouse schema to version 7.9.4 and migrate data.

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

To upgrade the data warehouse schema to version 7.9.4 and migrate data

1. Run the 792_UPGRADE_PRE_CTL_SCRIPT.sql script.
 - a. Open the SQL client for your database type, for example, SQLPLUS for Oracle, Query Analyzer for SQL Server, or a command window for DB2.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
 - c. Open the 792_UPGRADE_PRE_CTL_SCRIPT.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Run the ddl_794.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
3. Run the 792_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
4. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order indicated:
 UPGRADE_DIMENSIONS
 UPGRADE_FACTS
5. Verify the data migrated successfully by checking whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\SessLogs directory indicate errors or failures. Also, validate the data in the upgraded data warehouse.
6. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_79x tables, LKP_xxxx_x and 79x_XXXX_TMP.
 This step frees the space occupied by these backup tables.
7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

3.13 Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data

Follow this procedure to upgrade the data warehouse schema to version 7.9.5 and migrate data.

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

To upgrade the data warehouse schema to version 7.9.5 and migrate data

1. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
2. Run the `795_UPGRADE_PRE_CTL_SCRIPT.sql` script. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
3. Run the `UPGRADE_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the `ddl_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
6. Copy all of the domain value files in the folder `\OracleBI\dwrep\Informatica\LkpFiles` into the folder `<INFA_HOME>\server\infa_shared\LkpFiles`.
7. Migrate data into the upgraded data warehouse.
 - a. Copy the file `Upgrade.rep` file from the folder `OracleBI\dwrep\Upgrade\Informatica\Repository` into the folder `<INFA_HOME>\server\infa_shared\Backup`.
 - b. Launch the Informatica Administrator and restore `Upgrade.rep` (located in `<INFA_HOME>\server\infa_shared\Backup`).
 - c. Configure the Informatica parameter file `795_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 - d. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select `Connections`, and then select `Relational`), and edit the connect string, user and password for the relational connections as follows: create one relational connection based on the appropriate database platform for your OLTP database. Create the connection with the name `PARAM_OLTP`.

Edit the `PARAM_OLTP` connection to match your OLTP environment.

Edit the `PARAM_OLAP` connection to match your OLAP environment.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

- e. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_SBL and execute the UPGRADE_DIMENSIONS workflow.
- 8. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
 - c. Check whether the script 795_UPGRADE_PRE_DIMENSION_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log file for the ddl_795.ctl and Upgrade_795.ctl, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

- 9. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

- 10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

3.14 Upgrading the Data Warehouse Schema to Version 7.9.5.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.5.1.

To upgrade the data warehouse schema

- Use the DDLimp utility to run the ddl_7951.ctl script, which is located in the OracleBI\dwrep\Upgrade\CTLFiles folder. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

3.15 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data

You need to perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, or 7.9.5.1.

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

This section includes the following procedures:

- [Section 3.15.1, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1"](#)
- [Section 3.15.2, "Migrating Data Into the Upgraded Data Warehouse"](#)

- [Section 3.15.3, "Verifying the Data Migrated Successfully"](#)

3.15.1 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

Note: Upgrading to version 7.9.6.1 may involve running scripts that have a "796_" prefix.

To upgrade the data warehouse schema to version 7.9.6.1 and migrate data

1. If you are using an Oracle database, run the 796_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
 - c. Open the 796_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. If you are using a SQL Server or DB2 database, run the 7961_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
 - c. Open the 7961_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
3. If you are using a SQL Server database, run the 7961_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_796.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7961_UPGRADE_PRE_CTL_SCRIPT.sql:
 - a. Go to OracleBI\dwrep\Upgrade\DbScripts\<database type> folder, and open the 7961_UPGRADE_PRE_CTL_SCRIPT.sql file.
 - b. Search for the Datasource_Num_ID parameter, and make sure the value is correct for your existing implementation.
 - c. If you are using a DB2 database, search for the text "TBS_32K," and replace it with the appropriate 32K table space name that you have configured.
 - d. Execute the script.
6. Run the ddl_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
7. Run the 796_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)

9. Copy all of the domain value files in the folder
 \OracleBI\dwrep\Informatica\LkpFiles into the folder <INFA_HOME>\server\infa_shared\LkpFiles.

3.15.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Copy the file Upgrade.rep file from the folder
 OracleBI\dwrep\Upgrade\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.
2. Launch the Informatica Administrator and restore Upgrade.rep (located in
 <INFA_HOME>\server\infa_shared\Backup).
3. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For
 instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
4. Configure parameters specific to Siebel source systems. For instructions, see
[Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
5. In Informatica Workflow Manager, open the Relational Connection Browser (in the
 menu bar, select Connections, and then select Relational), and edit the connect
 string, user and password for the relational connections as follows:
 - a. Edit the connection PARAM_OLTP_SIEBEL to match your OLTP environment.
 - b. Edit the connection PARAM_OLAP to match your OLAP environment.
 - c. Edit the connection PARAM_DAC to match your DAC database.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

6. Back up and truncate the table W_POSITION_DH.
7. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order indicated:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
9. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For

instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)

11. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
12. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
14. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\

3.15.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961_UPGRADE_DROP_INDEXES.sql

7961_UPGRADE_PRE_CTL_SCRIPT.sql

7961_UPGRADE_PRE_DIMENSION.sql

7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the ddl_7961.ctl and Upgrade_7961.ctl scripts that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x, 79x_xxxx_TMP, and TMP_xxxx_79x.

This step frees the space occupied by these backup tables.

3.16 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1

Perform the procedures in this section if you are upgrading from Oracle BI Applications release 7.9.6 to release 7.9.6.1.

Note: This procedure is applicable only if you are using an Oracle database for your data warehouse database.

This section includes the following procedures:

- [Section 3.16.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 3.16.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 3.16.3, "Verifying the Data Migrated Successfully"](#)

3.16.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

To upgrade the data warehouse schema to version 7.9.6.1

1. Run the UPGRADE_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the 796ORA_TO_7961ORA_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
3. Run the ddl_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

3.16.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
4. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
5. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
6. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

3.16.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961_UPGRADE_DROP_INDEXES.sql

7961_UPGRADE_PRE_CTL_SCRIPT.sql

7961_UPGRADE_PRE_DIMENSION.sql

7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the ddl_7961.ctf and Upgrade_7961.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

3.17 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data

This section contains the following procedures:

- [Section 3.17.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 3.17.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 3.17.3, "Verifying the Data Migrated Successfully"](#)

3.17.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

To upgrade the data warehouse schema to version 7.9.6.2

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the 7962_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.

4. Run the UPGRADE_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7962_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the ddl_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

3.17.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7962_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Section D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
- 3. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
- 4. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
- 5. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Section D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
- 7. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

3.17.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql

7962_UPGRADE_PRE_CTL_SCRIPT.sql

7962_UPGRADE_PRE_DIMENSION.sql

createETLIndexes.sql

createQueryIndexes.sql

4. Check the log files for the ddl_7962.ctf and Upgrade_7962.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables.

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, then click OK.

3.18 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Section 3.18.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.3"](#)
- [Section 3.18.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 3.18.3, "Verifying the Data Migrated Successfully"](#)

3.18.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

3. If you are using a SQL Server database, run the 7963_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
6. Run the DDL_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

3.18.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7963_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
2. Run the 7963_UPGRADE_DROP_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
3. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
4. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
5. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
6. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
7. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.

If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
8. Run the 7963_UPGRADE_CREATE_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example,

Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

10. Run the createQueryIndices.sql script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

3.18.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7963_UPGRADE_PRE_CTL_SCRIPT.sql

7963_UPGRADE_PRE_DIMENSION.sql

7963_UPGRADE_DROP_ETLINDICES.sql

7963_UPGRADE_CREATE_ETLINDICES.sql

dropQueryIndices.sql

createQueryIndices.sql

4. Check the log files for the ddl_7963.ctl and Upgrade_7963.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.

- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

3.19 Localizing the Policy Store

The Oracle BI Applications 7.9.6.3 installer contains different policy stores for different languages. The default policy store that is deployed during the installation is in English. If you want to configure the policy store for any of the other available languages, perform the procedure below.

To configure the policy store for languages other than English

1. Stop the Oracle BI Services.
2. Copy the system-jazn-data_<LN>.xml file from \$ORACLE_BI_HOME\biapps\admin\provisioning\localization\ to \$DOMAIN_HOME\config\fmwconfig\

For example, to implement the policy store for French, you would copy the file system-jazn-data_fr.xml into the \fmwconfig directory.
3. Back up the existing system-jazn-data.xml file in \$DOMAIN_HOME\config\fmwconfig\

4. Rename \$DOMAIN_HOME\config\fmwconfig\system-jazn-data_<LN>.xml to system-jazn-data.xml.
5. Start Oracle BI Services.

3.20 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server (embedded LDAP server).

To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, myrealm).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).
4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

3.21 Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and web catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Siebel Analytics Repository version 7.5.x. This repository is the one that you originally received from Oracle as part of your

current Siebel Analytics 7.5.x installation. Standard repositories from previous releases are available in the installation folder \$ORACLE_BI_HOME\biapps\upgrade.

- The Siebel Analytics Repository you are currently using (the repository deployed in production).
- The Web Catalog you are currently using (the Web Catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

Note: If your Siebel Analytics Repository and Web Catalog are in a Siebel Analytics format, you must first upgrade them to the Oracle BI Enterprise Edition 10g format before you can upgrade them to the 11g format. For more information, see the *Oracle Business Intelligence Infrastructure Upgrade Guide Version 10.1.3.2*, which is located on the Oracle Technology Network in the Oracle Business Intelligence Suite Enterprise Edition Documentation Library Version 10.1.3.4.1.

Note: Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 3.19, "Localizing the Policy Store."](#)

3.22 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

Note: Before you begin this procedure, you must have already upgraded your Siebel Analytics repository to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 3.21, "Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

This process merges your customizations of a prior release of the Siebel Analytics repository with the new version of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository

To upgrade the repository, perform the following tasks:

- [Section 3.22.1, "Preparing for the Oracle BI Repository Upgrade"](#)

- [Section 3.22.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 3.22.3, "Comparing the Oracle BI Repositories"](#)
- [Section 3.22.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 3–1](#) provides the names and descriptions of the repositories used in the examples in this section.

Table 3–1 Names of Analytics Repositories used in Examples

Name of Repository	Description
SiebelAnalytics_7x.rpd	The standard Siebel Analytics repository for the version you are upgrading from. Note: Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade.
OracleBIAnalyticsApps.rpd	The standard Oracle BI repository for the version you are upgrading to.
Customer_SiebelAnalyticsApps.rpd	The Siebel Analytics repository that contains your customizations for the version you are upgrading from.
Merged_Repository_OracleBI.rpd	The Oracle BI repository that contains your customizations for the version you are upgrading to.

3.22.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

To prepare for the Analytics repository upgrade

1. Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
 - Original
 - AfterEqualize
 - AfterMerge
 - AfterManualWork
 - AfterRegressions
2. Copy the following repositories to the folder \OracleBIUpgrade\Original:
 - The standard (original) repository that you upgraded to the 11g format in [Section 3.21, "Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version"](#) (for example, SiebelAnalytics.rpd).
 - The production repository that you also upgraded to the 11g format in [Section 3.21](#) (for example, Customer_SiebelAnalyticsApps.rpd).
 - The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Siebel Analytics for one or more modules using a Siebel Analytics repository in which you extracted the corresponding projects for the modules from the standard Siebel Analytics repository file you received from the previous release, you need to extract the same projects from the SiebelAnalytics_7x.rpd file and use this as your original

repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Siebel Analytics repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the original SiebelAnalytics_7x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

3.22.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories (for example, siebelAnalyticsApps_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer_SiebelAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer_SiebelAnalyticsApps.rpd whether it is coming from the SiebelAnalyticsApps_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W_DAY_D" might be considered equivalent to "Core"."W DAY D" if the characters "_" and " " have been declared as equivalent.
- **Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the equalizerpds executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the \$ORACLE_BI_HOME\biapps\upgrade folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

Table 3–2 provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

Table 3–2 Rename MAP Files to Be Used for Various Releases

Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)	Rename MAP File to Be Used
Siebel Business Analytics Applications 7.0.x	Not available
Siebel Business Analytics Applications 7.5.x	Not available
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)	Rename77-7963.map

Table 3–2 (Cont.) Rename MAP Files to Be Used for Various Releases

Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)	Rename MAP File to Be Used
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)	Rename771-7963.map
Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release	Rename782-7963.map
Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release	Rename783-7963.map
Oracle BI Applications 7.9.0	Rename79x-7963.map
Oracle BI Applications 7.9.1	Rename79x-7963.map
Oracle BI Applications 7.9.2	Rename79x-7963.map
Oracle BI Applications 7.9.3	Rename793to7963.map
Oracle BI Applications 7.9.4	Rename794to7963.map
Oracle BI Applications 7.9.5	Rename79x-7963.map
Oracle BI Applications 7.9.5.1	Rename7951to7963.map
Oracle BI Applications 7.9.5.2	Rename7951to7963.map
Oracle BI Applications 7.9.6	Rename79x-7963.map
Oracle BI Applications 7.9.6.2	Not required.

The syntax of the `equalizerpds` command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udml_utf8_file_name_equalization] [-O output_repository_name]
[-Y equalStringSet]
-Y          Treat the characters as equals.
/?          Display this usage information and exit.
```

To equalize a repository

1. Copy the appropriate MAP file from the `$ORACLE_BI_HOME\biapps\upgrade` folder into the folder where you will execute `equalizerpds.exe`, for example, `\OracleBIUpgrade\Original`.
2. Run `equalizerpds.exe` to equalize the repository from the latest installation (for example, `OracleBIAnalyticsApps.rpd`) with the original repository (for example, `SiebelAnalyticsAppss_7x.rpd`). An example of the `equalizerpds` command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\SiebelAnalytics_7x.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the `$ORACLE_BI_HOME\biapps\upgrade` folder.

If the `equalizerpds.exe` executable file runs correctly, no errors are returned.

3. Run `equalizerpds.exe` to equalize your customized repository (for example, `Customer_SiebelAnalytics.rpd`) with the original repository (for example, `SiebelAnalytics_79x.rpd`). An example of the `equalizerpds` command is as follows:

```

equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\SiebelAnalytics_7x.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_SiebelAnalytics.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_SiebelAnalytics.rpd

```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.

Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

3.22.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3.22.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, SiebelAnalytics_7x.rpd, OracleBIAnalyticsApps.rpd, and Customer_SiebelAnalytics.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, SiebelAnalytics_7x.rpd).
6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.

9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, `Merged_Repository_OracleBIAnalyticsApps.rpd`.
The new repository will contain the final results of the merge.
10. Select the **Equalize during merge** check box.
11. Click **Next**.
12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.
When all rows have a value in the Decision field, the Finish button is enabled.
14. Select the **Check consistency of the merged RPD** check box.
15. Click **Finish**.
This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.
16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.
You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.
Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.
You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```
17. Copy the repository to the folder `\OracleBIUpgrade\AfterManualWork`.

3.23 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 3.21, "Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

You need to perform the following tasks to complete the presentation merge process.

- [Section 3.23.1, "Trimming the Input Presentation Catalog"](#)

- [Section 3.23.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 3.23.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This is the unmodified presentation catalog that you received with the Oracle BI Applications release that you are upgrading from.
- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This is the presentation catalog currently in use at your organization.

Caution: In releases of Oracle BI Applications previous to 7.9, the Oracle BI Presentation Catalog (formerly known as the Siebel Analytics Web Catalog) was stored in a single file rather than in a directory structure of individual files. If you have a previous version of the Presentation Catalog, you will need to convert it to the new format. For more information about how to convert the Presentation Catalog to the new format, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*

3.23.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

3.23.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 3.21, "Upgrading the Siebel Analytics Repository and Web Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.

For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.
3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name> -oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group -new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

Note: If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

3.23.3 Testing the Results of the Presentation Catalog Upgrade

Note: Before you perform this step, you must first migrate the data into the upgraded data warehouse.

The Presentation Catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the Presentation Catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged Presentation Catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new Presentation Catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

3.24 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the presentation catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE_BI_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.
5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAnalyticsApps.rpd and the production repository.

Upgrading Siebel Business Analytics 7.7, 7.7.x and 7.8.x

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

This section contains instructions for upgrading Oracle's Siebel Business Analytics releases 7.7, 7.7.x, and 7.8.x.

This section includes the following topics:

- [Section 4.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 4.2, "Upgrading Oracle BI Applications"](#)
- [Section 4.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 4.4, "Upgrading the Informatica Repository"](#)
- [Section 4.5, "Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC"](#)
- [Section 4.6, "Upgrading and Configuring DAC"](#)
- [Section 4.7, "Overview of Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases"](#)
- [Section 4.8, "Upgrading the Data Warehouse Schema to Version 7.9.0 for Non-Teradata Databases"](#)
- [Section 4.9, "Importing New Schema Definitions Into the Siebel Transactional Database"](#)
- [Section 4.10, "Migrating Data into the Data Warehouse Upgraded to Version 7.9.0 for Non-Teradata Databases"](#)

- Section 4.11, "Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data for Non-Teradata Databases"
- Section 4.12, "Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data for Non-Teradata Databases"
- Section 4.13, "Upgrading the Data Warehouse Schema to Version 7.9.5.1 for Non-Teradata Databases"
- Section 4.14, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data for Non-Teradata Databases"
- Section 4.15, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases"
- Section 4.16, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases"
- Section 4.17, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases"
- Section 4.18, "Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases"
- Section 4.19, "Localizing the Policy Store"
- Section 4.20, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"
- Section 4.21, "Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version"
- Section 4.22, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"
- Section 4.23, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog"
- Section 4.24, "Regression Testing the Oracle BI Repository Merge"

4.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

Note: If you are using a Siebel Business Analytics platform, you must first upgrade the platform to Oracle BI Infrastructure version 10.1.3.4.1, and then upgrade the platform to Oracle BI Enterprise Edition 11g (11.1.1.5.0).

For instructions on upgrading a Siebel Business Analytics platform to Oracle BI Infrastructure version 10.1.3.4.1, see the *Oracle Business Intelligence Infrastructure Upgrade Guide Version 10.1.3.2*, which is located on the Oracle Technology Network in the Oracle Business Intelligence Suite Enterprise Edition Documentation Library Version 10.1.3.4.1.

4.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release. This will be the backup of your DAC Client and Server and the DAC metadata files.)

4.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ significantly from previous versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 4.4, "Upgrading the Informatica Repository."](#)

In [Section 4.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle_BI_DW_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle_BI_DW_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

4.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

Note: This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 4.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle_BI_DW_Base repository that you restore in the steps below.

2. Copy the Oracle_BI_DW_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.

Note: The Oracle_BI_DW_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 4.2, "Upgrading Oracle BI Applications."](#)

3. Restore the Oracle_BI_DW_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
 - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
 - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.

- c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

4.5 Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

4.6 Upgrading and Configuring DAC

This section includes information you must follow to upgrade and configure DAC.

This section includes the following topics:

- [Section 4.6.1, "Installing the DAC Platform and Oracle BI Applications Metadata Repository Files"](#)
- [Section 4.6.2, "Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica"](#)
- [Section 4.6.3, "Upgrading the DAC Repository for Siebel Business Analytics"](#)

4.6.1 Installing the DAC Platform and Oracle BI Applications Metadata Repository Files

Note: You must retain your current DAC environment (Siebel Business Analytics 7.7, 7.7.x, or 7.8.x), including the DAC Client and Server and Repository. This is necessary because you cannot open the version 7.7, 7.7.x or 7.8.x DAC Repository with DAC version 10.1.3.4.1.

The current version of DAC is installed by its own installer and not the Oracle BI Applications installer. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

After you install the current version of DAC, you then need to copy metadata files from the machine hosting Oracle BI Applications to the machines hosting the DAC Client and Server. You then need to import the new metadata into the DAC Repository. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Installing and Setting Up the DAC Platform

Note: You must perform all of the procedures in this section, including installing JDBC drivers, creating ODBC database connections, and installing pmrep and pmcmd command line programs.

- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

4.6.2 Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica

You need to perform certain configuration tasks to enable the DAC Client and Server to work with Oracle BI Applications and Informatica PowerCenter. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring the SiebelUnicodeDB Custom Property

Note: This procedure is required only if your source to target data movement configuration is Unicode to Unicode.

- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

4.6.3 Upgrading the DAC Repository for Siebel Business Analytics

This section applies to upgrades of Siebel Business Analytics 7.7, 7.7.x, and 7.8.x.

Note: For Siebel Business Analytics 7.8.x, if your DAC Repository customizations are limited and you can identify them, use the procedure in this section, which is a manual procedure. If your customizations are extensive and you cannot identify them, you can use the procedure in [Section 5.6.3, "Upgrading the DAC Repository,"](#) which is an automated process that uses the DAC version 10.1.3.4.1 Upgrade/Merge Wizard.

Before you begin this procedure, do the following:

- Determine what customizations were made to your existing DAC Repository. Moving the customizations in your existing DAC Repository into the new DAC Repository is a manual process.

- Make sure you have renamed and backed up your existing DAC Repository into a different database. When you backup the DAC Repository, you export the DAC metadata, in XML format (using the DAC's Export tool), into a folder other than the standard DAC export folder where backups are stored (DAC\export). For instructions on exporting DAC metadata, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.
- Make sure you have set up the DAC system properties, Informatica PowerCenter Services, and database connections according to the instructions in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

In the procedure below you will manually move the customizations in the existing DAC Repository into the new, default version 7.9.6.2 DAC Repository.

To manually move your customizations from the previous DAC Repository into the default version 7.9.6.3 DAC Repository

1. Make a copy of the default source system container. (You cannot change the metadata in the default container. You must make a copy of it in order to be able to modify the objects it contains.) For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.
2. In the copy of the source system container, import the custom data warehouse tables:
 - a. Navigate to the Tables tab in the Design view.
 - b. Right-click and select Import from Database, and then click Import Database Tables.
 - c. In the Data Sources drop-down list, select DataWarehouse.
 - d. Click Read Tables.
 - e. Select the tables you want to import, and then click Import Tables.
3. Import the custom columns for the tables you imported in Step 2:
 - a. In the Tables tab, query for the tables you imported in Step 2.
 - b. Right-click and select Import from Database, and then click Import Database Columns.
 - c. In the Importing Columns... dialog box, select "All records in the list," and click OK.
 - d. In the Data Sources drop-down list, select DataWarehouse.
 - e. Click Read Columns.
 - f. Select the columns you want to import, and then click Import Columns.
 - g. Navigate to the Tables tab, and then click the Columns child tab, and add the Foreign Key to Table and Foreign Key to Column attributes for the newly imported columns.
4. Import the indices for the custom data warehouse tables.
 - a. In the Tables tab, query for the tables you imported in Step 2.
 - b. Right-click and select Import Indices.
 - c. In the Importing Indices... dialog box, select "All records in the list," and click OK.
 - d. In the Data Sources drop-down list, select DataWarehouse.

- e. Click Read Indices.
 - f. Select the indices you want to import, and then click Import Indices.
5. Import the custom source tables:
 - a. Navigate to the Tables tab in the Design view.
 - b. Right-click and select Import from Database, and then click Import Database Tables.
 - c. In the Data Sources drop-down list, select the appropriate source.
 - d. Click Read Tables.
 - e. Select the tables you want to import, and then click Import Tables.

You do not have to import columns for the custom source tables. Columns for source tables are not required.
6. Import columns for any standard data warehouse tables that were extended, and add the appropriate attributes in the DAC.
7. Create new custom logical and physical task folders.
 - a. In the DAC, from the Tools menu, select Seed Data, then select Task Folders.
 - b. To create a custom logical folder, click New.
 - c. In the Name field, enter a name for the custom logical folder, for example, Custom Logical.
 - d. In the Type field, select Logical.
 - e. To create a custom physical folder, click New.
 - f. In the Name field, enter a name for the custom physical folder, for example, Custom Physical.
 - g. In the Type field, select Physical.
8. Register the folders you created in Step 7 in the Source System Folders tab.
 - a. In the Design view, select the Source System Folders tab.
 - b. Click New.
 - c. Enter the name of the Custom Logical folder in the Logical Folder field.
 - d. Enter the name of the Custom Physical folder in the Physical Folder field, and save the record.
9. Modify the task attributes for workflows in the custom folder that are modified standard mappings, that is, standard (out-of-the-box) mappings from the previous release that you copied into the Informatica custom folder.
 - a. Navigate to the Tasks tab and query for the Informatica workflow names that are in the Informatica custom folder under Command for Incremental Load or Command for Full Load.

You must review the workflows in the custom folder in Informatica Workflow Manager.
 - b. For each task, change the Folder Name (in the Edit child tab) to the Custom Logical folder name.
 - c. For each task, right-click and select Synchronize Tasks.

- d. In the Task Synchronization dialog box, select Selected Record Only, and click OK.
This step adds source and target tables to the task.
 - e. Click Yes in the Synchronizing Task(s)... dialog box to proceed.
An informational message will indicate the results of the process.
 - f. Click OK.
10. Add tasks that were created as new tasks in the current implementation.
- a. In the Design view, select the Tasks tab, and then select the Edit child tab.
 - b. For each new task, copy the names of the Informatica workflows into the fields Command for Incremental Load and Command for Full Load.
 - c. Assign the appropriate values for the remaining fields in the Edit child tab.
 - d. In the Tasks tab, query for the tasks you entered in Step b.
 - e. Right-click the list of query results, and in the Task Synchronization dialog box, select "All records in the list," and click OK.
This step adds source tables and target tables to the task.
 - f. Click Yes in the Synchronizing Task(s)... dialog box to proceed.
An informational message will indicate the results of the process.
 - g. Click OK.

Note: If your customizations included new fact tables, you will need to create and assemble new subject areas as well as create and build new execution plans. If your customizations included extending dimension tables, you will need to reassemble your existing subject areas and rebuild your existing execution plans. You will also need to set the appropriate execution plan attributes, such as Prune Days. For instructions, see *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

4.7 Overview of Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases

The process for upgrading the data warehouse schema and migrating data for Siebel Business Analytics releases 7.7, 7.7.x, and 7.8.x involves multiple phases, as described below.

Note: The Oracle BI Applications upgrade logic is coded such that you must complete each of the phases listed below. To upgrade to the current release of Oracle BI Applications, you cannot skip any of the phases.

■ Phase 1

In phase 1, you will perform the following steps:

1. Upgrade the data warehouse schema to version 7.9.0. For instructions, see [Section 4.8](#).

2. Import new schema definitions into the Siebel transactional database. For instructions, see [Section 4.9](#).
 3. Migrate data into the data warehouse upgraded to version 7.9.0. For instructions, see [Section 4.10](#).
- **Phase 2**
In phase 2, you will upgrade your data warehouse schema to version 7.9.4 and migrate data into the upgraded data warehouse. For instructions, see [Section 4.11](#).
 - **Phase 3**
In phase 3, you will upgrade your data warehouse schema to version 7.9.5 and migrate data into the upgraded data warehouse. For instructions, see [Section 4.12](#).
 - **Phase 4**
In phase 4, you will upgrade your data warehouse schema to version 7.9.5.1. For instructions, see [Section 4.13](#).
 - **Phase 5**
In phase 5, you will do one of the following:
 - Upgrade your data warehouse schema from version 7.9.5.1 to version 7.9.6.1 and migrate data. For instructions, see [Section 4.14](#).
 - If you are currently running Oracle BI Applications version 7.9.6 and using an Oracle database, you can upgrade your data warehouse schema to version 7.9.6.1 and migrate data. For instructions, see [Section 4.15](#).
 - **Phase 6**
In phase 6, you will upgrade your data warehouse schema to version 7.9.6.2 and migrate data. For instructions, see [Section 4.16](#).
 - **Phase 7**
In phase 7, you will upgrade your data warehouse schema to version 7.9.6.3 and migrate data. For instructions, see [Section 4.17](#).

4.8 Upgrading the Data Warehouse Schema to Version 7.9.0 for Non-Teradata Databases

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

To upgrade the data warehouse schema

1. Run the UPGRADE.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the 790_UPGRADE_PRE_CTL_SCRIPT.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
 - c. Open the 790_UPGRADE_PRE_CTL_SCRIPT.sql file, and copy the contents into the SQL client.
 - d. Execute the script.

3. Run the DW.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the 790_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

4.9 Importing New Schema Definitions Into the Siebel Transactional Database

This procedure upgrades transactional database objects that relate to Oracle Business Intelligence Applications. It does not upgrade transactional database objects for Siebel CRM applications.

To import new schema definitions into the Siebel transactional database

1. Import schema definitions for non-image tables:
 - a. Access the DDLimp utility.
 - b. Run the DDL_OLTP.ctl script that is located in the OracleBI\dwrep\Upgrade\CTLFiles folder.

Note: If you receive an error message stating a particular object already exists in the database, use the DDLimp Merge argument (/M Y) in the DDLimp command to resolve the error.
2. Import schema definitions for image tables:
 - a. In DAC, go to the Design view, and select your custom container from the drop-down list to the right of the Execute button.
 - b. Click the Tables tab.
 - c. Query for all tables for which the image suffix is not null.
 - d. Right-click in the list of tables returned by the query, and select Change Capture Scripts, and then Generate Image and Trigger Scripts.
 - e. In the Triggers and Image Tables dialog box, do the following:
 - Select the option All Tables in the List.
 - Select the option Generate Image Table Scripts.
 - Select the appropriate database type for the source system.
 - Click OK.
 - f. Open the SQL client for the source system database type, for example, SQL Plus for Oracle, Query Analyzer for SQL Server, or a command window for DB2.

The script may contain many lines; therefore, you can save the script file as a SQL file and execute it in a SQL client.
 - g. Copy the scripts generated by DAC into the SQL client and execute them.

4.9.1 Verifying the Siebel Transactional Database Upgrade

Follow this procedure to verify the following tables were created in the Siebel transactional database.

To verify the transactional database upgrade

- For all upgrade paths, verify the following tables were created in the Siebel transactional database:
 - S_ETL_R_IMG_XXX
 - S_ETL_I_IMG_XXX
 - S_ETL_D_IMG_XXX
 - S_ETL_PARAM
 - S_ETL_PRD_ATTR
 - S_ETL_PRD_REL

4.10 Migrating Data into the Data Warehouse Upgraded to Version 7.9.0 for Non-Teradata Databases

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Copy the file Upgrade.rep file from the folder OracleBI\dwrep\Upgrade\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.
2. Launch the Informatica Administrator, and restore Upgrade.rep (located in <INFA_HOME>\server\infa_shared\Backup) into a database other than the database in which you restored Oracle_BI_DW_Base.rep.
3. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\SrcFiles and copy the *.csv files into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
4. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file <previous version of Siebel Analytics>_TENERIFE_UPG_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
5. Rename the file <previous version of Siebel Analytics>_TENERIFE_UPG_PARAMS.txt to TENERIFE_UPG_PARAMS.txt.
6. In the file TENERIFE_UPG_PARAMS.txt:
 - a. Search for the parameter \$\$SourceConnection.
 - b. Set the value to match your Siebel applications (OLTP) version:
 - * SEBL_63
 - * SEBL_753
 - * SEBL_771
 - * SEBL_78
 - * SEBL_80 (use this value for OLTP versions 8.x and 8.1.1)
 - * SEBL_VERT_753
 - * SEBL_VERT_771
 - * SEBL_VERT_78
 - * SEBL_VERT_80 (use this value for OLTP versions 8.x and 8.1.1)

For example, if your Siebel applications (OLTP) version is Siebel Industry Applications 7.8, the parameter should appear as:

```
$$SourceConnection=SEBL_VERT_78
```

7. In the file TENERIFE_UPG_PARAMS.txt:
 - a. Search for the parameter \$\$Source_Container.
 - b. Set the value to match your Siebel applications (OLTP) version:
 - * Siebel 6.3
 - * Siebel 7.5.3
 - * Siebel 7.7.1
 - * Siebel 7.8
 - * Siebel 8.0 (use this value for OLTP versions 8.x and 8.1.1)
 - * Siebel 7.5.3 Vertical
 - * Siebel 7.7.1 Vertical
 - * Siebel 7.8 Vertical
 - * Siebel 8.0 Vertical (use this value for OLTP versions 8.x and 8.1.1)

For example, if your Siebel applications (OLTP) version is Siebel Industry Applications 7.8, the parameter should appear as:

```
$$Source_Container=Siebel 7.8 Vertical
```

8. In the file TENERIFE_UPG_PARAMS.txt, edit the ETL_PROC_WID parameter as follows:

```
MPLT_GET_ETL_PROC_WID.$$ETL_PROC_WID=<latest ETL_PROC_WID value from your database>
```

You can get this value from W_PARAM_G.ETL_PROC_WID.

9. If you are running Siebel Industry Applications (Vertical), in the file TENERIFE_UPG_PARAMS.txt, set the VERTICAL_UPGRADE parameter to 1. For example:


```
$$VERTICAL_UPGRADE=1
```

10. In the Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and edit the connect string, user and password for the relational connections as follows:
 - a. Edit the PARAM_OLTP_SIEBEL connection to point to the Siebel transactional database.
 - b. Edit the DataWarehouse connection to point to the newly upgraded data warehouse database.
 - c. Create or edit the PARAM_DAC_OLD connection to point to the previous DAC Repository database (the version from which you are upgrading).
 - d. Create or edit the PARAM_DAC_NEW connection to point to the DAC Repository database.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

11. If you are upgrading from Siebel Industry Applications (Vertical), define the alignment rule to be used for ETL loads.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\SrcFiles and copy the file AlignmentType_LS_782.csv into the SrcFiles folder on the Informatica Server machine, for example, <INFA_HOME>\server\infa_shared\SrcFiles.
 - b. Rename the file AlignmentType_LS_782.csv to AlignmentType.csv.
 - c. Open the file AlignmentType.csv, and enter one of the following alignment item types under the ALIGN_TYPE field:
 - * Zipcode
 - * Account/Contact
 - * Brick
 - d. Save the file.

The Oracle Business Analytics Warehouse allows only one alignment type to be used for ETL loads during upgrade.

12. In Informatica Workflow Manager, navigate to the appropriate folder and execute the workflows in the order specified:

Application Version	Folder Name	Workflow
7.7 to 7.9	UPGRADE_770_to_79	<ul style="list-style-type: none"> ■ MARKETING_LOAD ■ Upgrade_Dimensions_Industry ■ Update_Dimensions ■ Update_Dimension_Unspecified ■ Update_Facts ■ Upgrade_LS_Dimensions ■ Upgrade_LS_Facts ■ SIL_PositionDimensionHierarchy_Full ■ Load_INT_ORG_DH ■ DIMENSION_LOAD <p>Note: If you are using the SCD version of the dimension, replace the corresponding TENN_UPG_W_XXX_D_784_To_W_XXX_D session with TENN_UPG_W_XXX_D_784_SCD_To_W_XXX_D. This will upgrade the data from the W_XXX_SCD version of the dimension to the new SCD-enabled W_XXX_D dimension.</p> <ul style="list-style-type: none"> ■ DIMENSION_UNSPECIFIED_UPDATE ■ FACT_UPDATE ■ DAC_Metadata_Upgrade_Workflow

Application Version	Folder Name	Workflow
7.7.1.x to 7.9	UPGRADE_771_to_79	<ul style="list-style-type: none"> ■ Update_Dimensions ■ Update_Dimension_Unspecified ■ Update_Facts ■ Upgrade_LS_Dimensions ■ Upgrade_LS_Facts ■ SIL_PositionDimensionHierarchy_Full ■ Load_INT_ORG_DH ■ DIMENSION_LOAD <p>Note: If you are using the SCD version of the dimension, replace the corresponding TENN_UPG_W_XXX_D_784_To_W_XXX_D session with TENN_UPG_W_XXX_D_784_SCD_To_W_XXX_D. This will upgrade the data from the W_XXX_SCD version of the dimension to the new SCD-enabled W_XXX_D dimension.</p> <ul style="list-style-type: none"> ■ DIMENSION_UNSPECIFIED_UPDATE ■ FACT_UPDATE ■ DAC_Metadata_Upgrade_Workflow
7.8 to 7.9	UPGRADE_78_to_79	<ul style="list-style-type: none"> ■ SIL_PositionDimensionHierarchy_Full ■ Load_INT_ORG_DH ■ DIMENSION_LOAD <p>Note: If you are using the SCD version of the dimension, replace the corresponding TENN_UPG_W_XXX_D_784_To_W_XXX_D session with TENN_UPG_W_XXX_D_784_SCD_To_W_XXX_D. This will upgrade the data from W_XXX_SCD version of the dimension to the new SCD enabled W_XXX_D dimension.</p> <ul style="list-style-type: none"> ■ DIMENSION_UNSPECIFIED_UPDATE ■ FACT_UPDATE ■ DAC_Metadata_Upgrade_Workflow

13. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, navigate to the folder UPGRADE_790_to_791_SBL80UPG and run the following workflows in the order specified:
 - a. UPGRADE_DIMENSIONS
 - b. UPGRADE_FACTS
14. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
15. To verify the data migrated successfully:
 - a. Validate the data in the upgraded data warehouse.

- b. Check whether any of the Informatica mapping log files stored in the directory <INFA_HOME>\server\infa_shared\SessLogs indicates errors or failures.
- c. Check whether the SQL scripts (790_UPGRADE_PRE_CTL_SCRIPT.sql and 790_UPGRADE_PRE_DIMENSION_SCRIPT.sql) that you ran in the SQL client of the database failed or errored out while executing.
- d. Check the log files for the CTL files (Upgrade.ctl and DW.ctl) that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables.

- 16. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_784 tables, LKP_xxxx_x, and W_ASSET_D_TMP.

This step frees the space occupied by these backup tables.

4.11 Upgrading the Data Warehouse Schema to Version 7.9.4 and Migrating Data for Non-Teradata Databases

Follow this procedure to upgrade the data warehouse schema to version 7.9.4 and migrate data.

To upgrade the data warehouse schema and migrate data

1. Run the 792_UPGRADE_PRE_CTL_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
2. Run the ddl_794.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
3. Run the 792_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
4. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 UPGRADE_DIMENSIONS
 UPGRADE_FACTS
5. Verify the data migrated successfully by checking whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\SessLogs directory indicate errors or failures. Also, validate the data in the upgraded data warehouse.
6. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_79x tables, LKP_xxxx_x and 79x_XXXX_TMP.

This step frees the space occupied by these backup tables.

7. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

4.12 Upgrading the Data Warehouse Schema to Version 7.9.5 and Migrating Data for Non-Teradata Databases

Follow this procedure to upgrade the data warehouse schema to version 7.9.5 and migrate data.

To upgrade the data warehouse schema and migrate data

1. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
2. Run the `795_UPGRADE_PRE_CTL_SCRIPT.sql` script. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
3. Run the `UPGRADE_795.ctf` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the `ddl_795.ctf` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
6. Copy all of the domain value files in the folder `\OracleBI\dwrep\Informatica\LkpFiles` into the folder `<INFA_HOME>\server\infa_shared\LkpFiles`.
7. Migrate data into the upgraded data warehouse.
 - a. Copy the file `Upgrade.rep` file from the folder `OracleBI\dwrep\Upgrade\Informatica\Repository` into the folder `<INFA_HOME>\server\infa_shared\Backup`.
 - b. Launch the Informatica Administrator, and restore `Upgrade.rep` (located in `<INFA_HOME>\server\infa_shared\Backup`).
 - c. Configure the Informatica parameter file `795_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 - d. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select `Connections`, and then select `Relational`), and edit the connect string, user and password for the relational connections as follows: create one relational connection based on the appropriate database platform for your OLTP database. Create the connection with the name `PARAM_OLTP`.
 Edit `PARAM_OLTP` connection to match your OLTP environment.
 Edit the `PARAM_OLAP` connection to match your OLAP environment.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

- e. In Informatica Workflow Manager, navigate to the folder `UPGRADE_794_to_795_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
8. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the `<INFA_HOME>\server\infa_shared\Sesslogs` directory indicates errors or failures.
 - c. Check whether the script `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log file for `ddl_795` and script `Upgrade_795.ctl` scripts, which you ran using the `DDLimp` command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

9. If the data migration was successful, drop the tables that were created during the upgrade process, such as `W_XXXX_X_79X`, `LKP_XXXX_X` and `79X_XXXX_TMP`.

This step frees the space occupied by these backup tables.
10. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

4.13 Upgrading the Data Warehouse Schema to Version 7.9.5.1 for Non-Teradata Databases

Follow this procedure to upgrade the data warehouse schema to version 7.9.5.1.

To upgrade the data warehouse schema

- Use the `DDLimp` utility to run the `ddl_7951.ctl` script, which is located in the `OracleBI\dwrep\Upgrade\CTLFiles` folder. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

4.14 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data for Non-Teradata Databases

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

Note: You can upgrade directly from version 7.9.5.1 to 7.9.6.1.

This section includes the following procedures:

- [Section 4.14.1, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1"](#)
- [Section 4.14.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 4.14.3, "Verifying the Data Migrated Successfully"](#)

4.14.1 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

Note: Upgrading to version 7.9.6.1 may involve running scripts that have a "796_" prefix.

To upgrade the data warehouse schema to version 7.9.6.1 and migrate data

1. If you are using an Oracle database, run the 796_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
 - c. Open the 796_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. If you are using a SQL Server or DB2 database, run the 7961_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
 - c. Open the 7961_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
3. If you are using a SQL Server database, run the 7961_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_796.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7961_UPGRADE_PRE_CTL_SCRIPT.sql:
 - a. Go to OracleBI\dwrep\Upgrade\DbScripts\<database type> folder, and open the 7961_UPGRADE_PRE_CTL_SCRIPT.sql file.
 - b. Search for the Datasource_Num_ID parameter, and make sure the value is correct for your existing implementation.

- c. If you are using a DB2 database, search for the text "TBS_32K," and replace it with the appropriate 32K table space name that you have configured.
- d. Execute the script.
6. Run the ddl_7961.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
7. Run the 796_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
- 9. Copy all of the domain value files in the folder \OracleBI\dwrep\Informatica\LkpFiles into the folder <INFA_HOME>\server\infa_shared\LkpFiles.

4.14.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Copy the file Upgrade.rep file from the folder OracleBI\dwrep\Upgrade\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.
2. Launch the Informatica Administrator, and restore Upgrade.rep (located in <INFA_HOME>\server\infa_shared\Backup).
3. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
4. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
5. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and edit the connect string, user and password for the relational connections as follows:
 - a. Edit the connection PARAM_OLTP_SIEBEL to match your OLTP environment.
 - b. Edit the connection PARAM_OLAP to match your OLAP environment.
 - c. Edit the connection PARAM_DAC to match your DAC database.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

6. Back up and truncate the table W_POSITION_DH.
7. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS

- UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
 9. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\ - 10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
 - 11. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
 - 12. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 - 13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
 - 14. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\

4.14.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
 - 7961_UPGRADE_DROP_INDEXES.sql
 - 7961_UPGRADE_PRE_CTL_SCRIPT.sql
 - 7961_UPGRADE_PRE_DIMENSION.sql
 - 7961_UPGRADE_POST_SCRIPT.sql
4. Check the log files for the ddl_7961.ctl and Upgrade_7961.ctl scripts that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as `W_xxxx_x_79x`, `LKP_xxxx_x`, `79x_xxxx_TMP`, and `TMP_xxxx_79x`.

This step frees the space occupied by these backup tables.

4.15 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6 to release 7.9.6.1.

Note: This procedure is applicable only if you are using an Oracle database for your data warehouse database.

This section includes the following procedures:

- [Section 4.15.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 4.15.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 4.15.3, "Verifying the Data Migrated Successfully"](#)

4.15.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

To upgrade the data warehouse schema to version 7.9.6.1

1. Run the `UPGRADE_7961.ctf` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the `796ORA_TO_7961ORA_UPGRADE_PRE_CTL_SCRIPT.sql`. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\Oracle`.
3. Run the `ddl_7961.ctf` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

4.15.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the `7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Oracle`.
2. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)

3. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
4. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
5. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
6. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

4.15.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961_UPGRADE_DROP_INDEXES.sql

7961_UPGRADE_PRE_CTL_SCRIPT.sql

7961_UPGRADE_PRE_DIMENSION.sql

7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the ddl_7961.ctl and Upgrade_7961.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

4.16 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases

This section contains the following procedures:

- [Section 4.16.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 4.16.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 4.16.3, "Verifying the Data Migrated Successfully"](#)

4.16.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

To upgrade the data warehouse schema to version 7.9.6.2

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the 7962_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7962_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
6. Run the ddl_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

4.16.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7962_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
4. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
5. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Section D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
7. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

4.16.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql

7962_UPGRADE_PRE_CTL_SCRIPT.sql

7962_UPGRADE_PRE_DIMENSION.sql

createETLIndexes.sql

createQueryIndexes.sql

4. Check the log files for the ddl_7962.ctl and Upgrade_7962.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables.

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, then click OK.

4.17 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Section 4.17.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.3"](#)
- [Section 4.17.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 4.17.3, "Verifying the Data Migrated Successfully"](#)

4.17.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
Note: You must drop the query indexes before running the scripts to upgrade the schema.
3. If you are using a SQL Server database, run the 7963_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
6. Run the DDL_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

4.17.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7963_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
2. Run the 7963_UPGRADE_DROP_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
3. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
4. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
5. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
6. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
7. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE  
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
```

```
LIKE ' %'OR CITY LIKE ' % 'OR CITY LIKE ' %'OR ZIPCODE LIKE ' % 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
8. Run the 7963_UPGRADE_CREATE_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
- 10. Run the createQueryIndices script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

4.17.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
7963_UPGRADE_PRE_CTL_SCRIPT.sql
7963_UPGRADE_PRE_DIMENSION.sql
7963_UPGRADE_DROP_ETLINDICES.sql
7963_UPGRADE_CREATE_ETLINDICES.sql
dropQueryIndices.sql
createQueryIndices.sql
4. Check the log files for the ddl_7963.ctl and Upgrade_7963.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.18 Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases

If your data warehouse database platform is Teradata, you can upgrade to Oracle BI Applications version 7.9.6.3 from Oracle BI Applications 7.9.x.

This section includes the following topics:

- [Section 4.18.1, "Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3"](#)
- [Section 4.18.2, "Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3"](#)
- [Section 4.18.3, "Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3"](#)
- [Section 4.18.4, "Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3"](#)
- [Section 4.18.5, "Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3"](#)
- [Section 4.18.6, "Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3"](#)

4.18.1 Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.0 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 795_UPGRADE_TMP_TABLES.sql
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 794_RENAME_AND_EXCEPTIONS.sql script.

- a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 794_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 794_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
 4. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:
 - Select the appropriate physical data source name for the Teradata database.
 - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 795_UPDATE_DEFAULTS.sql
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Execute the 795_Backup Tables.sql script.
 7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 8. Configure the Informatica parameter file 795_UPG_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 10. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_SBL and execute the UPGRADE_DIMENSIONS workflow.

11. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the `7951_Backup Tables.sql` script.
13. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Section D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
14. Configure the Informatica parameter file `7951_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
15. In Informatica Workflow Manager, navigate to the folder `UPGRADE_795_to_7951_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
16. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the `796_Backup Tables.sql` script.
18. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Configure the Informatica parameter file `796_UPG_PARAMS.txt` file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
20. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
21. Back up and truncate the table `W_POSITION_DH`.
22. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7951_to_796_SBL` and execute the following workflows in the order specified:
 - `UPGRADE_DIMENSIONS`
 - `UPGRADE_FACTS`

- RESET_DAC_REFRESH_DATES
23. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 24. Execute the 7961_Backup_Tables.sql script.
 25. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 26. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
 27. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 28. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 29. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 30. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 31. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)

32. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
33. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
34. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
35. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
37. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
38. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
39. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
40. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
41. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, then click OK.

4.18.2 Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.4 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 795_UPGRADE_TMP_TABLES.sql
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLESsq.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 795_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 795_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.

- e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 795_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
 4. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 795_UPDATE_DEFAULTS.sql
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Execute the 795_Backup Tables.sql script.
 7. Configure the Informatica parameter file 795_UPG_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 10. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_SBL and execute the UPGRADE_DIMENSIONS workflow.
 11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 12. Execute the 7951_Backup Tables.sql script.

13. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
14. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
15. In Informatica Workflow Manager, navigate to the folder UPGRADE_795_to_7951_SBL and execute the UPGRADE_DIMENSIONS workflow.
16. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the 796_Backup Tables.sql script.
18. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
20. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
21. Back up and truncate the table W_POSITION_DH.
22. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
23. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the 7961_Backup Tables.sql script.
 25. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 26. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
 27. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 28. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 29. Execute the 7962_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 30. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 31. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Section A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
 32. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
 33. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
34. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
35. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
37. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
38. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
39. Upgrade geography-related attributes if necessary for your environment:
- a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
40. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

41. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.18.3 Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.5 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the

teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.

4. Generate upgrade scripts from DAC.

a. Log into DAC.

b. From the menu bar, select Tools, then ETL Management, and then Configure.

c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.

i. Execute the scripts in the upgrade-regular.sql file.

j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.

k. Recreate STATISTICS on any column on which they were dropped.

5. Execute the following scripts:
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
6. Execute the 7951_Backup Tables.sql script.
7. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
10. In Informatica Workflow Manager, navigate to the folder UPGRADE_795_to_7951_SBL and execute the UPGRADE_DIMENSIONS workflow.
11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the 796_Backup Tables.sql script.
13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)

14. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
15. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
16. Back up and truncate the table W_POSITION_DH.
17. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
18. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Execute the 7961_Backup Tables.sql script.
20. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
21. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
22. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
23. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the 7962_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.

25. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
27. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
28. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the `7963_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
30. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
32. If you are using Siebel Industry Applications, you need to set an additional parameter in `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
33. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_SBL` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
34. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```


- b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.
If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
35. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.18.4 Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.5.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 7961_RENAME_AND_EXCEPTIONS.sql script.

- c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
 4. Generate upgrade scripts from DAC.
 - a. Log into DAC.
 - b. From the menu bar, select Tools, then ETL Management, and then Configure.
 - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.
 - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.
 - e. Do the following:
 - Select the appropriate physical data source name for the Teradata database.
 - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Execute the 796_Backup Tables.sql script.
 7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 8. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
 9. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
 10. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 11. Back up and truncate the table W_POSITION_DH.
 12. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS

- RESET_DAC_REFRESH_DATES
- 13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 14. Execute the 7961_Backup_Tables.sql script.
- 15. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 16. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
- 17. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 18. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 19. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 20. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 21. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)

22. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
23. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
25. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
27. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
28. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
29. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
30. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.18.5 Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 7962_UPGRDE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.

3. The `teradata.missing.defaults.properties` file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the `teradata.missing.defaults.properties` file to match your environment. The file is located in `\bifoundation\dac\conf\sqlgen\sql\teradata`.

4. Generate upgrade scripts from DAC.

a. Log into DAC.

b. From the menu bar, select Tools, then ETL Management, and then Configure.

c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in `\bifoundation\DAC\conf\sqlgen\sql\teradata`:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the `upgrade-regular.sql` file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the `upgrade-questionable.sql` file. It is for information purposes only.

f. Open the `upgrade-regular.sql` and `upgrade-questionable.sql` files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.

i. Execute the scripts in the `upgrade-regular.sql` file.

- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 7. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 8. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
 9. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
 10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 11. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 12. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 13. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
 14. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)

15. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
16. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\.

If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
17. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
18. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.18.6 Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.2 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the 7963_UPGRADE_TMP_TABLES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the 7963_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.

2. The `teradata.missing.defaults.properties` file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the `teradata.missing.defaults.properties` file to match your environment. The file is located in `\bifoundation\dac\conf\sqlgen\sql\teradata`.

3. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in `\bifoundation\DAC\conf\sqlgen\sql\teradata`:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the `upgrade-regular.sql` file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file `upgrade-questionable.sql`. It is for information purposes only.

- f. Open the `upgrade-regular.sql` and `upgrade-questionable.sql` files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
- i. Execute the scripts in the `upgrade-regular.sql` file.

- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the 7963_UPDATE_DEFAULTS.sql scripts.
5. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
7. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
8. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
9. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
10. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.

- a. In the Informatica Administrator, select the Upgrade repository service.
- b. In the General Properties area of the Properties tab, click Edit.
- c. Make sure the operating mode of the repository service is set to Exclusive.
- d. Click OK.
- e. Choose Actions, and then click Delete Contents.
- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

4.19 Localizing the Policy Store

The Oracle BI Applications 7.9.6.3 installer contains different policy stores for different languages. The default policy store that is deployed during the installation is in English. If you want to configure the policy store for any of the other available languages, perform the procedure below.

To configure the policy store for languages other than English

1. Stop the Oracle BI Services.
2. Copy the system-jazn-data_<LN>.xml file from \$ORACLE_BI_HOME\biapps\admin\provisioning\localization\ to \$DOMAIN_HOME\config\fmwconfig\

For example, to implement the policy store for French, you would copy the file system-jazn-data_fr.xml into the \fmwconfig directory.
3. Back up the existing system-jazn-data.xml file in \$DOMAIN_HOME\config\fmwconfig\

4. Rename \$DOMAIN_HOME\config\fmwconfig\system-jazn-data_<LN>.xml to system-jazn-data.xml.
5. Start Oracle BI Services.

4.20 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server embedded LDAP server.

To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, myrealm).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).
4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

4.21 Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and web catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Siebel Analytics Repository version 7.7, 7.7.x, or 7.8.x. This repository is the one that you originally received from Oracle as part of your current Siebel Analytics 7.7, 7.7.x, or 7.8.x installation. Standard repositories from previous releases are available in the installation folder \$ORACLE_BI_HOME\biapps\upgrade.
- The Siebel Analytics Repository you are currently using (the repository deployed in production).
- The Web Catalog you are currently using (the Web Catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

Note: If your Siebel Analytics Repository and Web Catalog are in a Siebel Analytics format, you must first upgrade them to the Oracle BI Enterprise Edition 10g format before you can upgrade them to the 11g format. For more information, see the *Oracle Business Intelligence Infrastructure Upgrade Guide Version 10.1.3.2*, which is located on the Oracle Technology Network in the Oracle Business Intelligence Suite Enterprise Edition Documentation Library Version 10.1.3.4.1.

Note: Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 4.19, "Localizing the Policy Store."](#)

4.22 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

Note: Before you begin this procedure, you must have already upgraded your Siebel Analytics repository to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 4.21, "Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

This process merges your customizations of a prior release of the Siebel Analytics repository with the new version of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository

To upgrade the repository, perform the following tasks:

- [Section 4.22.1, "Preparing for the Oracle BI Repository Upgrade"](#)
- [Section 4.22.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 4.22.3, "Comparing the Oracle BI Repositories"](#)
- [Section 4.22.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 4–1](#) provides the names and descriptions of the repositories used in the examples in this section.

Table 4–1 Names of Analytics Repositories used in Examples

Name of Repository	Description
SiebelAnalytics_7x.rpd	The standard Siebel Analytics repository for the version you are upgrading from. Note: Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade.
OracleBIAnalyticsApps.rpd	The standard Oracle BI repository for the version you are upgrading to.
Customer_SiebelAnalyticsApps.rpd	The Siebel Analytics repository that contains your customizations for the version you are upgrading from.
Merged_Repository_OracleBI.rpd	The Oracle BI repository that contains your customizations for the version you are upgrading to.

4.22.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

To prepare for the Analytics repository upgrade

1. Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
 - Original
 - AfterEqualize
 - AfterMerge
 - AfterManualWork

- AfterRegressions
2. Copy the following repositories to the folder \OracleBIUpgrade\Original:
 - The standard (original) repository that you upgraded to the 11g format in [Section 4.21, "Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version"](#) (for example, SiebelAnalytics.rpd).
 - The production repository that you also upgraded to the 11g format in [Section 4.21](#) (for example, Customer_SiebelAnalyticsApps.rpd).
 - The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Siebel Analytics for one or more modules using a Siebel Analytics repository in which you extracted the corresponding projects for the modules from the standard Siebel Analytics repository file you received from the previous release, you need to extract the same projects from the SiebelAnalytics_7x.rpd file and use this as your original repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Siebel Analytics repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the original SiebelAnalytics_7x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

4.22.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories (for example, siebelAnalyticsApps_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer_SiebelAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer_SiebelAnalyticsApps.rpd whether it is coming from the SiebelAnalyticsApps_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W_DAY_D" might be considered equivalent to "Core"."W DAY D" if the characters "_" and " " have been declared as equivalent.

- Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the `equalizerpds` executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the `$ORACLE_BI_HOME\biapps\upgrade` folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

[Table 4–2](#) provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

Table 4–2 Rename MAP Files to Be Used for Various Releases

Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)	Rename MAP File to Be Used
Siebel Business Analytics Applications 7.0.x	Not available
Siebel Business Analytics Applications 7.5.x	Not available
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)	Rename77-7963.map
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)	Rename771-7963.map
Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release	Rename782-7963.map
Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release	Rename783-7963.map
Oracle BI Applications 7.9.0	Rename79x-7963.map
Oracle BI Applications 7.9.1	Rename79x-7963.map
Oracle BI Applications 7.9.2	Rename79x-7963.map
Oracle BI Applications 7.9.3	Rename793to7963.map
Oracle BI Applications 7.9.4	Rename794to7963.map
Oracle BI Applications 7.9.5	Rename79x-7963.map
Oracle BI Applications 7.9.5.1	Rename7951to7963.map
Oracle BI Applications 7.9.5.2	Rename7951to7963.map
Oracle BI Applications 7.9.6	Rename79x-7963.map
Oracle BI Applications 7.9.6.2	Not required

The syntax of the `equalizerpds` command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udm1_utf8_file_name_equalization] [-O output_repository_name]
[-Y equalStringSet]
-Y          Treat the characters as equals.
/?          Display this usage information and exit.
```

To equalize a repository

- Copy the appropriate MAP file from the `$ORACLE_BI_HOME\biapps\upgrade` folder into the folder where you will execute `equalizerpds.exe`, for example, `\OracleBIUpgrade\Original`.
- Run `equalizerpds.exe` to equalize the repository from the latest installation (for example, `OracleBIAnalyticsApps.rpd`) with the original repository (for example,

SiebelAnalyticsAppss_7x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\SiebelAnalytics_7x.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the \$ORACLE_BI_HOME\biapps\upgrade folder.

If the equalizerpds.exe executable file runs correctly, no errors are returned.

3. Run equalizerpds.exe to equalize your customized repository (for example, Customer_SiebelAnalytics.rpd) with the original repository (for example, SiebelAnalytics_79x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\SiebelAnalytics_7x.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_SiebelAnalytics.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_SiebelAnalytics.rpd
```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.

Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

4.22.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4.22.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, SiebelAnalytics_7x.rpd, OracleBIAnalyticsApps.rpd, and Customer_SiebelAnalytics.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, SiebelAnalytics_7x.rpd).
6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.
9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, Merged_Repository_OracleBIAnalyticsApps.rpd.
The new repository will contain the final results of the merge.
10. Select the **Equalize during merge** check box.
11. Click **Next**.
12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.
When all rows have a value in the Decision field, the Finish button is enabled.
14. Select the **Check consistency of the merged RPD** check box.
15. Click **Finish**.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

17. Copy the repository to the folder \OracleBIUpgrade\AfterManualWork.

4.23 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 4.21, "Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

You need to perform the following tasks to complete the presentation catalog merge process.

- [Section 4.23.1, "Trimming the Input Presentation Catalog"](#)
- [Section 4.23.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 4.23.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This is the unmodified presentation catalog that you received with the Oracle BI Applications release that you are upgrading from.
- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This is the presentation catalog currently in use at your organization.

4.23.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

4.23.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 4.21](#), "Upgrading the Siebel Analytics Repository and the Web Catalog to the Oracle BI Enterprise Edition 11g Version."

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.

For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.
3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name>
-oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group
-new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

Note: If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4.23.3 Testing the Results of the Presentation Catalog Upgrade

Note: Before you perform this step, you must have already migrated the data into the upgraded data warehouse.

The Presentation Catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the Presentation Catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged Presentation Catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new Presentation Catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

4.24 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the Presentation Catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE_BI_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.

5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAnalyticsApps.rpd and the production repository.

Upgrading Oracle BI Applications 7.9.x for Siebel Source Systems

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

This section contains instructions for upgrading Oracle BI Applications releases when you are running Siebel CRM Applications as your source system.

This section includes the following topics:

- [Section 5.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 5.2, "Upgrading Oracle BI Applications"](#)
- [Section 5.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 5.4, "Upgrading the Informatica Repository"](#)
- [Section 5.5, "Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC"](#)
- [Section 5.6, "Upgrading and Configuring DAC"](#)
- [Section 5.7, "Importing New Schema Definitions Into the Siebel Transactional Database"](#)
- [Section 5.8, "Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases"](#)
- [Section 5.9, "Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases"](#)
- [Section 5.10, "Localizing the Policy Store"](#)

- Section 5.11, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"
- Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"
- Section 5.13, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"
- Section 5.14, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog"
- Section 5.15, "Regression Testing the Oracle BI Repository Merge"

5.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

5.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release. This will be the backup of your DAC Client and Server and the DAC metadata files.)

5.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ significantly from previous versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 5.4, "Upgrading the Informatica Repository."](#)

In [Section 5.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle_BI_DW_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle_BI_DW_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

5.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

Note: This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 5.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle_BI_DW_Base repository that you restore in the steps below.

2. Copy the Oracle_BI_DW_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.

Note: The Oracle_BI_DW_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 5.2, "Upgrading Oracle BI Applications."](#)

3. Restore the Oracle_BI_DW_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
 - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
 - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.
 - c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

5.5 Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

5.6 Upgrading and Configuring DAC

This section includes information you must follow to upgrade and configure DAC.

This section includes the following topics:

- [Section 5.6.1, "Installing the DAC Platform and Oracle BI Applications Metadata Repository Files"](#)
- [Section 5.6.2, "Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica"](#)

- [Section 5.6.3, "Upgrading the DAC Repository"](#)

5.6.1 Installing the DAC Platform and Oracle BI Applications Metadata Repository Files

The current version of DAC is installed by its own installer and not the Oracle BI Applications installer. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

After you install DAC, you then need to copy metadata files from the machine hosting Oracle BI Applications to the machines hosting the DAC Client and Server. You then need to import the new metadata into the DAC Repository. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Installing and Setting Up the DAC Platform

Note: You must perform all of the procedures in this section, including installing JDBC drivers, creating ODBC database connections, and installing pmrep and pmcmd command line programs.

- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

5.6.2 Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica

You need to perform certain configuration tasks to enable the DAC Client and Server to work with Oracle BI Applications and Informatica PowerCenter. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring the SiebelUnicodeDB Custom Property

Note: This procedure is required only if your source to target data movement configuration is Unicode to Unicode.

- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

5.6.3 Upgrading the DAC Repository

You will use the Refresh Base option of the DAC's Upgrade/Merge Wizard to complete the upgrade of your existing DAC Repository. For information about how to

use the Refresh Base option of the Upgrade/Merge Wizard, see the topic titled, "About the Refresh Base Option," in "Chapter 10: Upgrading, Comparing and Merging DAC Repositories," in the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

The *Oracle Business Intelligence Data Warehouse Administration Console User's Guide* is available in the Oracle Business Intelligence Data Warehouse Administration Console Documentation Library on the Oracle Technology Network.

Before you upgrade the DAC Repository, do the following:

- Make a copy of your default source system container. (You cannot change the metadata in the preconfigured container. You must make a copy of it in order to be able to modify the objects it contains.) For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

5.7 Importing New Schema Definitions Into the Siebel Transactional Database

This procedure upgrades transactional database objects that relate to Oracle Business Intelligence Applications. It does not upgrade transactional database objects for Siebel CRM applications.

To import new schema definitions into the Siebel transactional database

1. Import schema definitions for non-image tables:
 - a. Access the DDLimp utility.
 - b. Run the DDL_OLTP.ctf script that is located in the OracleBI\dwrep\Upgrade\CTLFiles folder.

Note: If you receive an error message stating a particular object already exists in the database, use the DDLimp Merge argument (/M Y) in the DDLimp command to resolve the error.
2. Import schema definitions for image tables:
 - a. In DAC, go to the Design view, and select your custom container from the drop-down list to the right of the Execute button.
 - b. Click the Tables tab.
 - c. Query for all tables for which the image suffix is not null.
 - d. Right-click in the list of tables returned by the query, and select Change Capture Scripts, and then Generate Image and Trigger Scripts.
 - e. In the Triggers and Image Tables dialog box, do the following:
 - Select the option All Tables in the List.
 - Select the option Generate Image Table Scripts.
 - Select the appropriate database type for the source system.
 - Click OK.
 - f. Open the SQL client for the source system database type, for example, SQL Plus for Oracle, Query Analyzer for SQL Server, or a command window for DB2.

The script may contain many lines; therefore, you can save the script file as a SQL file and execute it in a SQL client.

- g. Copy the scripts generated by DAC into the SQL client and execute them.

5.7.1 Verifying the Siebel Transactional Database Upgrade

Follow this procedure to verify the following tables were created in the Siebel transactional database.

To verify the transactional database upgrade

- For all upgrade paths, verify the following tables were created in the Siebel transactional database:
 - S_ETL_R_IMG_XXX
 - S_ETL_I_IMG_XXX
 - S_ETL_D_IMG_XXX
 - S_ETL_PARAM
 - S_ETL_PRD_ATTR
 - S_ETL_PRD_REL

5.8 Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases

This section contains the following topics:

- [Section 5.8.1, "Upgrading the Data Warehouse Schema From Version 7.9.0 to 7.9.4 and Migrating Data"](#)
- [Section 5.8.2, "Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data"](#)
- [Section 5.8.3, "Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data"](#)
- [Section 5.8.4, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data"](#)
- [Section 5.8.5, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data"](#) (Applicable only if you are using an Oracle database as your data warehouse database.)
- [Section 5.8.6, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data"](#)
- [Section 5.8.7, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data"](#)

If you are currently running Oracle BI Applications release 7.9.0, 7.9.1, 7.9.2 or 7.9.3, you need to perform the procedures in [Section 5.8.1](#), [Section 5.8.2](#), and [Section 5.8.3](#), and [Section 5.8.4](#), [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.4, you need to perform the procedures in [Section 5.8.2](#), [Section 5.8.3](#), [Section 5.8.4](#), [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.5, you need to perform the procedures in [Section 5.8.3](#), [Section 5.8.4](#), [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.5.1, you need to perform the procedures in [Section 5.8.4](#), [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.6, you need to perform the procedures in [Section 5.8.5](#), [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.6.1, you need to perform the procedures in [Section 5.8.6](#), and [Section 5.8.7](#).

If you are currently running Oracle BI Applications release 7.9.6.2, you need to perform the procedures in [Section 5.8.7](#).

These procedures add new tables, columns, and indexes to the existing data warehouse schema. They also modify the existing data warehouse schema objects.

5.8.1 Upgrading the Data Warehouse Schema From Version 7.9.0 to 7.9.4 and Migrating Data

You need to perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1, 7.9.2 or 7.9.3.

To upgrade the data warehouse schema to version 7.9.4 and migrate data

1. Run the 792_UPGRADE_PRE_CTL_SCRIPT.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
 - c. Open the 792_UPGRADE_PRE_CTL_SCRIPT.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Run the ddl_794.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
3. Run the 792_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in the OracleBI\dwrep\Upgrade\DbScripts\<<database type> folder.
4. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified below:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
5. Verify the data migrated successfully by checking whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\SessLogs directory indicate errors or failures. Also, validate the data in the upgraded data warehouse.
6. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_79x tables, LKP_xxxx_x and 79x_XXXX_TMP. This step frees the space occupied by these backup tables.
7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

5.8.2 Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data

You need to perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1, 7.9.2, 7.9.3 or 7.9.4.

To upgrade the data warehouse schema to version 7.9.5 and migrate data

1. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
 2. Run the `795_UPGRADE_PRE_CTL_SCRIPT.sql` script. This file is located in the `OracleBI\dwrep\Upgrade\DbScripts\<database type>` folder.
 3. Run the `UPGRADE_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
 4. Run the `ddl_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
 5. Run the `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in the `OracleBI\dwrep\Upgrade\DbScripts\<database type>` folder.
 6. Copy all of the domain value files in the folder `\OracleBI\dwrep\Informatica\LkpFiles` into the folder `<INFA_HOME>\server\infa_shared\LkpFiles`.
 7. Migrate data into the upgraded data warehouse.
 - a. Copy the file `Upgrade.rep` file from the folder `OracleBI\dwrep\Upgrade\Informatica\Repository` into the folder `<INFA_HOME>\server\infa_shared\Backup`.
 - b. Launch the Informatica Administrator, and restore `Upgrade.rep` (located in `<INFA_HOME>\server\infa_shared\Backup`).
 - c. Configure the Informatica parameter file `795_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 - d. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select `Connections`, and then select `Relational`), and edit the connect string, user and password for the relational connections as follows:
 Edit the `PARAM_OLTP` connection to match your OLTP environment.
 Edit the `PARAM_OLAP` connection to match your OLAP environment.
-
- Note:** If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.
- If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.
-
- e. In Informatica Workflow Manager, navigate to the folder `UPGRADE_794_to_795_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
8. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.

- b. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
- c. Check whether the script 795_UPGRADE_PRE_DIMENSION_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
- d. Check the log files for DDL_795.ctl and Upgrade_795.ctl, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

- 9. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.
This step frees the space occupied by these backup tables.
- 10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

5.8.3 Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data

You need to perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4 or 7.9.5.

To upgrade the data warehouse schema to version 7.9.5.1

- Use the DDLimp utility to run the ddl_7951.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

5.8.4 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data

You need to perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5 or 7.9.5.1.

This procedure adds new tables, columns, and indexes to the existing data warehouse schema. It also modifies the existing data warehouse schema objects.

This section includes the following procedures:

- [Section 5.8.4.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 5.8.4.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 5.8.4.3, "Verifying the Data Migrated Successfully"](#)

5.8.4.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

Note: Upgrading to version 7.9.6.1 may involve running scripts that have a "796_" prefix.

To upgrade the data warehouse schema to version 7.9.6.1 and migrate data

1. If you are using an Oracle database, run the 796_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
 - c. Open the 796_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. If you are using a SQL Server or DB2 database, run the 7961_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
 - c. Open the 7961_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
3. If you are using a SQL Server database, run the 7961_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_796.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7961_UPGRADE_PRE_CTL_SCRIPT.sql:
 - a. Go to OracleBI\dwrep\Upgrade\DbScripts\<database type> folder, and open the 7961_UPGRADE_PRE_CTL_SCRIPT.sql file.
 - b. Search for the Datasource_Num_ID parameter, and make sure the value is correct for your existing implementation.
 - c. If you are using a DB2 database, search for the text "TBS_32K," and replace it with the appropriate 32K table space name that you have configured.
 - d. Execute the script.
6. Run the ddl_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
7. Run the 796_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
9. Copy all of the domain value files in the folder \OracleBI\dwrep\Informatica\LkpFiles into the folder <INFA_HOME>\server\infa_shared\LkpFiles.

5.8.4.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Copy the file Upgrade.rep file from the folder OracleBI\dwrep\Upgrade\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.
2. Launch the Informatica Administrator, and restore Upgrade.rep (located in <INFA_HOME>\server\infa_shared\Backup).
3. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
4. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
5. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and edit the connect string, user and password for the relational connections as follows:
 - a. Edit the connection PARAM_OLTP_SIEBEL to match your OLTP environment.
 - b. Edit the connection PARAM_OLAP to match your OLAP environment.
 - c. Edit the connection PARAM_DAC to match your DAC database.

Note: If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

6. Back up and truncate the table W_POSITION_DH.
7. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
9. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
11. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)

12. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
14. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\

5.8.4.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961_UPGRADE_DROP_INDEXES.sql

7961_UPGRADE_PRE_CTL_SCRIPT.sql

796 UPGRADE_PRE_DIMENSION.sql

7961_UPGRADE_PRE_DIMENSION.sql

7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the ddl_7961.ctl and Upgrade_796.ctl scripts that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x, 79x_xxxx_TMP, and TMP_xxxx_79x.

This step frees the space occupied by these backup tables.

5.8.5 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data

Perform the procedures in this section if you are upgrading from Oracle BI Applications release 7.9.6 to release 7.9.6.1.

Note: This procedure is applicable only if you are using an Oracle database for your data warehouse database.

This section includes the following procedures:

- [Section 5.8.5.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 5.8.5.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 5.8.5.3, "Verifying the Data Migrated Successfully"](#)

5.8.5.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

To upgrade the data warehouse schema to version 7.9.6.1

1. Run the UPGRADE_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the 796ORA_TO_7961ORA_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
3. Run the ddl_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

5.8.5.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
4. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
5. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
6. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

5.8.5.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.

2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
 - 7961_UPGRADE_DROP_INDEXES.sql
 - 7961_UPGRADE_PRE_CTL_SCRIPT.sql
 - 7961_UPGRADE_PRE_DIMENSION.sql
 - 7961_UPGRADE_POST_SCRIPT.sql
4. Check the log files for the ddl_7961.ctl and Upgrade_7961.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP. This step frees the space occupied by these backup tables.
6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.8.6 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data

This section contains the following procedures:

- [Section 5.8.6.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 5.8.6.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 5.8.6.3, "Verifying the Data Migrated Successfully"](#)

You need to complete each of these procedures in order to upgrade the data warehouse schema to version 7.9.6.2 and migrate data.

5.8.6.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

To upgrade the data warehouse schema to version 7.9.6.2

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the 7962_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7962_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the ddl_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

5.8.6.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7962_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository.
- 3. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
- 4. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
- 5. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
- 7. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

5.8.6.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql
 7962_UPGRADE_PRE_CTL_SCRIPT.sql
 7962_UPGRADE_PRE_DIMENSION.sql
 createETLIndexes.sql
 createQueryIndexes.sql

4. Check the log files for the ddl_7962.ctl and Upgrade_7962.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables.

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.8.7 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Section 5.8.7.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.3"](#)
- [Section 5.8.7.2, "Migrating Data Into the Upgraded Data Warehouse"](#)
- [Section 5.8.7.3, "Verifying the Data Migrated Successfully"](#)

5.8.7.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
Note: You must drop the query indexes before running the scripts to upgrade the schema.
3. If you are using a SQL Server database, run the 7963_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7963.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the DDL_7963.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

5.8.7.2 Migrating Data Into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7963_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Run the 7963_UPGRADE_DROP_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 3. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
- 4. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
- 5. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
- 6. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 7. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```


- b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
- 8. Run the 7963_UPGRADE_CREATE_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
- 10. Run the createQueryIndices.sql script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

5.8.7.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
7963_UPGRADE_PRE_CTL_SCRIPT.sql
7963_UPGRADE_PRE_DIMENSION.sql
7963_UPGRADE_DROP_ETLINDICES.sql
7963_UPGRADE_CREATE_ETLINDICES.sql
dropQueryIndices.sql
createQueryIndices.sql
4. Check the log files for the ddl_7963.ctf and Upgrade_7963.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9 Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases

If your data warehouse database platform is Teradata, you can upgrade to Oracle Business Intelligence Applications version 7.9.6.3 from Oracle Business Intelligence Applications 7.9.x.

This section includes the following topics:

- [Section 5.9.1, "Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3"](#)
- [Section 5.9.2, "Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3"](#)
- [Section 5.9.3, "Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3"](#)
- [Section 5.9.4, "Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3"](#)
- [Section 5.9.5, "Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3"](#)
- [Section 5.9.6, "Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3"](#)

5.9.1 Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.0 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 795_UPGRADE_TMP_TABLES.sql
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 794_RENAME_AND_EXCEPTIONS.sql script.

- a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 794_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 794_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
 4. Generate upgrade scripts from DAC.
 - a. Log into DAC.
 - b. From the menu bar, select Tools, then ETL Management, and then Configure.
 - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.
 - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.
 - e. Do the following:
 - Select the appropriate physical data source name for the Teradata database.
 - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 795_UPDATE_DEFAULTS.sql
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Execute the 795_Backup Tables.sql script.
 7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 8. Configure the Informatica parameter file 795_UPG_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 10. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_SBL and execute the UPGRADE_DIMENSIONS workflow.

11. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the `7951_Backup Tables.sql` script.
13. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository.
14. Configure the Informatica parameter file `7951_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
15. In Informatica Workflow Manager, navigate to the folder `UPGRADE_795_to_7951_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
16. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the `796_Backup Tables.sql` script.
18. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Configure the Informatica parameter file `796_UPG_PARAMS.txt` file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
20. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
21. Back up and truncate the table `W_POSITION_DH`.
22. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7951_to_796_SBL` and execute the following workflows in the order specified:
 - `UPGRADE_DIMENSIONS`
 - `UPGRADE_FACTS`
 - `RESET_DAC_REFRESH_DATES`

23. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the `7961_Backup Tables.sql` script.
25. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file `7961_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
27. In Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_SBL` and execute the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
28. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the `7962_Backup Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
30. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
32. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.

33. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
34. Execute the `7963_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
35. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Configure the Informatica parameter file `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
37. If you are using Siebel Industry Applications, you need to set an additional parameter in `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
38. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_SBL` and execute first the `UPGRADE_DIMENSIONS` workflow, and then the `UPGRADE_FACTS` workflow.
39. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the `7963_UPGRADE_POST_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.

If the query returns a count of zero, you do not need to run the `7963_UPGRADE_POST_SCRIPT.sql` script.
40. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 41. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9.2 Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.4 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 795_UPGRADE_TMP_TABLES.sql
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 795_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 795_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 795_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.

3. The `teradata.missing.defaults.properties` file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the `teradata.missing.defaults.properties` file to match your environment. The file is located in `\bifoundation\dac\conf\sqlgen\sql\teradata`.

4. Generate upgrade scripts from DAC.

a. Log into DAC.

b. From the menu bar, select Tools, then ETL Management, and then Configure.

c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in `\bifoundation\DAC\conf\sqlgen\sql\teradata`:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the `upgrade-regular.sql` file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the `upgrade-questionable.sql` file. It is for information purposes only.

f. Open the `upgrade-regular.sql` and `upgrade-questionable.sql` files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.

i. Execute the scripts in the `upgrade-regular.sql` file.

- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
- 5. Execute the following scripts:
 - 795_UPDATE_DEFAULTS.sql
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
- 6. Execute the 795_Backup Tables.sql script.
- 7. Configure the Informatica parameter file 795_UPG_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
- 8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
- 9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
- 10. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_SBL and execute the UPGRADE_DIMENSIONS workflow.
- 11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 12. Execute the 7951_Backup Tables.sql script.
- 13. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
- 14. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
15. In Informatica Workflow Manager, navigate to the folder UPGRADE_795_to_7951_SBL and execute the UPGRADE_DIMENSIONS workflow.
16. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the 796_Backup Tables.sql script.
18. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
20. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
21. Back up and truncate the table W_POSITION_DH.
22. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
23. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the 7961_Backup Tables.sql script.

25. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file `7961_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
27. In Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_SBL` and execute the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
28. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the `7962_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
30. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
32. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
33. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

34. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
35. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
37. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
38. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow, and then the UPGRADE_FACTS workflow.
39. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
40. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
41. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.

- e. Choose Actions, and then click Delete Contents.
- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9.3 Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.5 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 7951_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
 - a. Log into DAC.
 - b. From the menu bar, select Tools, then ETL Management, and then Configure.
 - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
- i. Execute the scripts in the upgrade-regular.sql file.
- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
- k. Recreate STATISTICS on any column on which they were dropped.

5. Execute the following scripts:

- 7951_UPDATE_DEFAULTS.sql
- 7961_UPDATE_DEFAULTS.sql
- 7962_UPDATE_DEFAULTS.sql
- 7963_UPDATE_DEFAULTS.sql

6. Execute the 7951_Backup Tables.sql script.

7. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
9. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 UPGRADE_DIMENSIONS
 UPGRADE_FACTS
10. In Informatica Workflow Manager, navigate to the folder UPGRADE_795_to_7951_SBL and execute the UPGRADE_DIMENSIONS workflow.
11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the 796_Backup Tables.sql script.
13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
14. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
15. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
16. Back up and truncate the table W_POSITION_DH.
17. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:

- UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
18. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 19. Execute the 7961_Backup Tables.sql script.
 20. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 21. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
 22. Back up and truncate the table W_POSITION_DH.
 23. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 24. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 25. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 26. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 27. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
- 28. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
- 29. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 30. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 31. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 32. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
- 33. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
- 34. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow, and then the UPGRADE_FACTS workflow.
- 35. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.

36. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
37. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9.4 Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.5.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - `7961_UPGRADE_TMP_TABLES.sql`
 - `7962_UPGRADE_TMP_TABLES.sql`
 - `7963_UPGRADE_TMP_TABLESsq`
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
 - c. Open the `<version>_UPGRADE_TMP_TABLES.sql` file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the `7961_RENAME_AND_EXCEPTIONS.sql` script.
 - a. Navigate to the folder `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
 - b. Open the `7961_RENAME_AND_EXCEPTIONS.sql` script.
 - c. Search for the following statements to locate columns with RENAME commands.


```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
 - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.

- e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 7961_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
6. Execute the 796_Backup Tables.sql script.
7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
8. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
9. Configure parameters specific to Siebel source systems. For instructions, see [Appendix A.2.2, "Setting Parameters in 796_UPG_PARAMS.txt for Siebel Source Systems."](#)
10. If you upgraded your transactional database to Siebel Applications 8.x or 8.1.1, run the workflows to migrate your data into the upgraded data warehouse.
 - a. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_TO_791_SBL80UPG.
 - b. Run the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
11. Back up and truncate the table W_POSITION_DH.
12. In Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_SBL and execute the following workflows in the order specified:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
 - RESET_DAC_REFRESH_DATES
13. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 14. Execute the 7961_Backup_Tables.sql script.
- 15. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 16. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
- 17. In Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_SBL and execute the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 18. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 19. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 20. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
- 21. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
- 22. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 23. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 24. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
25. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
 26. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
 27. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow, and then the UPGRADE_FACTS workflow.
 28. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
 29. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 30. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9.5 Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 7962_UPGRDE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.
 - a. Log into DAC.
 - b. From the menu bar, select Tools, then ETL Management, and then Configure.
 - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.
 - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.
 - e. Do the following:
 - Select the appropriate physical data source name for the Teradata database.
 - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the following scripts:
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 5. Execute the 7962_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 7. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
 8. In Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_SBL and execute the UPGRADE_DIMENSIONS workflow.
 9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 10. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.

11. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Configure the Informatica parameter file `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
13. If you are using Siebel Industry Applications, you need to set an additional parameter in `7963_UPG_PARAMS.txt`. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
14. In Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_SBL` and execute the `UPGRADE_DIMENSIONS` workflow.
15. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:


```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
 - b. If the query returns a count greater than zero, run the `7963_UPGRADE_POST_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.

If the query returns a count of zero, you do not need to run the `7963_UPGRADE_POST_SCRIPT.sql` script.
16. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.

- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.9.6 Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.2 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the 7963_UPGRADE_TMP_TABLES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the 7963_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the 7963_UPDATE_DEFAULTS.sql scripts.
 5. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 6. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 7. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
 8. If you are using Siebel Industry Applications, you need to set an additional parameter in 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.2.4, "Setting Parameters in 7963_UPG_PARAMS.txt for Siebel Industry Applications Source Systems."](#)
 9. In Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_SBL and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 10. Upgrade geography-related attributes if necessary for your environment:
 - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

5.10 Localizing the Policy Store

The Oracle BI Applications 7.9.6.3 installer contains different policy stores for different languages. The default policy store that is deployed during the installation is in English. If you want to configure the policy store for any of the other available languages, perform the procedure below.

To configure the policy store for languages other than English

1. Stop the Oracle BI Services.
2. Copy the system-jazn-data_<LN>.xml file from \$ORACLE_BI_HOME\biapps\admin\provisioning\localization\ to \$DOMAIN_HOME\config\fmwconfig\.

For example, to implement the policy store for French, you would copy the file system-jazn-data_fr.xml into the \fmwconfig directory.
3. Back up the existing system-jazn-data.xml file in \$DOMAIN_HOME\config\fmwconfig\.
4. Rename \$DOMAIN_HOME\config\fmwconfig\system-jazn-data_<LN>.xml to system-jazn-data.xml.
5. Start Oracle BI Services.

5.11 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server embedded LDAP server.

To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, `myrealm`).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).
4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

5.12 Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and web catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Oracle BI Repository. This repository is the one that you originally received from Oracle as part of your current Oracle BI Applications installation (the version from which you are upgrading). Standard repositories from previous releases are available in the installation folder `$ORACLE_BI_HOME\biapps\upgrade`.
- The Oracle BI Repository you are currently using (the repository deployed in production).
- The Presentation Catalog you are currently using (the catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

Note: Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 5.10, "Localizing the Policy Store."](#)

5.13 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

Note: Before you begin this procedure, you must have already upgraded the Oracle BI repository you are currently using and the standard (out-of-the-box) Oracle BI repository version 7.9.x (the version from which you are upgrading) to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

This process merges your customizations of a prior release of the Oracle BI repository with the new release of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository.

To upgrade the repository, perform the following tasks:

- [Section 5.13.1, "Preparing for the Oracle BI Repository Upgrade"](#)
- [Section 5.13.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 5.13.3, "Comparing the Oracle BI Repositories"](#)
- [Section 5.13.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 5–1](#) provides the names and descriptions of the repositories used in the examples in this section.

Table 5–1 Names of Analytics Repositories used in Examples

Name of Repository	Description
OracleBIAnalyticsApps_79x.rpd	The standard Oracle BI repository for the version you are upgrading from. Note: Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade.

Table 5–1 (Cont.) Names of Analytics Repositories used in Examples

Name of Repository	Description
OracleBIAnalyticsApps.rpd	The standard Oracle BI repository for the version you are upgrading to.
Customer_OracleBIAnalyticsApps.rpd	The Oracle BI repository that contains your customizations for the version you are upgrading from.
Merged_Repository_OracleBI.rpd	The Oracle BI repository that contains your customizations for the version you are upgrading to.

5.13.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

To prepare for the Analytics repository upgrade

- Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
 - Original
 - AfterEqualize
 - AfterMerge
 - AfterManualWork
 - AfterRegressions
- Copy the following repositories to the folder \OracleBIUpgrade\Original:
 - The standard repository that you upgraded to the 11g format in [Section 5.12](#), "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version" (for example, OracleBIAnalyticsApps_79x.rpd).
 - The production repository that you also upgraded to the 11g format in [Section 5.12](#) (for example, Customer_OracleBIAnalyticsApps.rpd).
 - The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Oracle BI Applications for one or more modules using a Oracle BI repository in which you extracted the corresponding projects for the modules from the standard Oracle BI repository file you received from the previous release, you need to extract the same projects from the OracleBIAnalyticsApps_79x.rpd file and use this as your original repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Oracle BI repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the OracleBIAnalyticsApps_79x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

5.13.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories

(for example, OracleBIAnalyticsApps_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer_OracleBIAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer_OracleBIAnalyticsApps.rpd whether it is coming from the OracleBIAnalyticsApps_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W_DAY_D" might be considered equivalent to "Core"."W DAY D" if the characters "_" and " " have been declared as equivalent.
- **Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the equalizerpds executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the \$ORACLE_BI_HOME\biapps\upgrade folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

[Table 5–2](#) provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

Table 5–2 Rename MAP Files to Be Used for Various Releases

Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)	Rename MAP File to Be Used
Siebel Business Analytics Applications 7.0.x	Not available
Siebel Business Analytics Applications 7.5.x	Not available
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)	Rename77-7963.map
Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)	Rename771-7963.map
Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release	Rename782-7963.map
Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release	Rename783-7963.map
Oracle BI Applications 7.9.0	Rename79x-7963.map
Oracle BI Applications 7.9.1	Rename79x-7963.map
Oracle BI Applications 7.9.2	Rename79x-7963.map
Oracle BI Applications 7.9.3	Rename793to7963.map
Oracle BI Applications 7.9.4	Rename794to7963.map

Table 5–2 (Cont.) Rename MAP Files to Be Used for Various Releases

Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)	Rename MAP File to Be Used
Oracle BI Applications 7.9.5	Rename79x-7963.map
Oracle BI Applications 7.9.5.1	Rename7951to7963.map
Oracle BI Applications 7.9.5.2	Rename7951to7963.map
Oracle BI Applications 7.9.6	Rename79x-7963.map
Oracle BI Applications 7.9.6.2	Not required

The syntax of the equalizerpds command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udml_utf8_file_name_equalization] [-O ouput_repository_name]
[-Y equalStringSet]
-Y          Treat the characters as equals.
/?          Display this usage information and exit.
```

To equalize a repository

1. Copy the appropriate MAP file from the \$ORACLE_BI_HOME\biapps\upgrade folder into the folder where you will execute equalizerpds.exe, for example, \OracleBIUpgrade\Original.
2. Run equalizerpds.exe to equalize the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsAppss_793.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the \$ORACLE_BI_HOME\biapps\upgrade folder.

If the equalizerpds.exe executable file runs correctly, no errors are returned.

3. Run equalizerpds.exe to equalize your customized repository (for example, Customer_OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsApps_79x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_OracleBIAnalyticsApps.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_OracleBIAnalyticsApps.rpd
```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.

Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

5.13.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

5.13.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, OracleBIAnalyticsApps_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer_OracleBIAnalyticsApps.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, OracleBIAnalyticsApps_79x.rpd).
6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.
9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, Merged_Repository_OracleBIAnalyticsApps.rpd.
The new repository will contain the final results of the merge.
10. Select the **Equalize during merge** check box.
11. Click **Next**.
12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.

When all rows have a value in the Decision field, the Finish button is enabled.

14. Select the **Check consistency of the merged RPD** check box.

15. Click **Finish**.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

17. Copy the repository to the folder \OracleBIUpgrade\AfterManualWork.

5.14 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

You need to perform the following tasks to complete the presentation merge process.

- [Section 5.14.1, "Trimming the Input Presentation Catalog"](#)
- [Section 5.14.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 5.14.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This catalog is the output from the step you performed in [Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the version of the unmodified presentation catalog you received with the Oracle BI Applications release you are upgrading from that you upgraded to the 11g format.

- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This catalog is the output from the step you performed in [Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the presentation catalog currently in use at your organization that you upgraded to the 11g format.

5.14.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

5.14.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

Note: Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 5.12, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.
For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.
2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.
3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name>
-oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group
-new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

Note: If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

5.14.3 Testing the Results of the Presentation Catalog Upgrade

Note: Before you perform this step, you must first migrate the data into the upgraded data warehouse.

The Presentation Catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the Presentation Catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged Presentation Catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new Presentation Catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

5.15 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the Presentation Catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE_BI_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.
5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAnalyticsApps.rpd and the production repository.

Part II

Upgrading When Your Source System Is Oracle EBS

Part II contains instructions for upgrading to the current release of Oracle BI Applications from previous releases when you are running Oracle EBS as your source system.

Part II contains [Chapter 6, "Upgrading Oracle BI Applications for Oracle Source Systems."](#)

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

Upgrading Oracle BI Applications for Oracle Source Systems

Note: Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

This section contains instructions for upgrading Oracle BI Applications when you are running Oracle EBS as your source system.

This section includes the following topics:

- [Section 6.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 6.2, "Upgrading Oracle BI Applications"](#)
- [Section 6.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 6.4, "Upgrading the Informatica Repository"](#)
- [Section 6.5, "Configuring Informatica PowerCenter Version 9.0.1 Hotfix 2 to Work With Oracle BI Applications and DAC"](#)
- [Section 6.6, "Upgrading and Configuring DAC"](#)
- [Section 6.7, "Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases"](#)
- [Section 6.8, "Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases"](#)
- [Section 6.9, "Localizing the Policy Store"](#)
- [Section 6.10, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"](#)

- [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#)
- [Section 6.12, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"](#)
- [Section 6.13, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 6.14, "Regression Testing the Oracle BI Repository Merge"](#)

6.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

6.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release. This will be the backup of your DAC Client and Server and the DAC metadata files.)

6.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ from previous PowerCenter versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 6.4, "Upgrading the Informatica Repository."](#)

In [Section 6.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle_BI_DW_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle_BI_DW_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

6.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the PowerCenter Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

Note: This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 6.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle_BI_DW_Base repository that you restore in the steps below.

2. Copy the Oracle_BI_DW_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.

Note: The Oracle_BI_DW_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 6.2, "Upgrading Oracle BI Applications."](#)

3. Restore the Oracle_BI_DW_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
 - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
 - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.
 - c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

6.5 Configuring Informatica PowerCenter Version 9.0.1 Hotfix 2 to Work With Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 Hotfix 2 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

6.6 Upgrading and Configuring DAC

This section includes information you must follow to upgrade and configure DAC.

This section includes the following topics:

- [Section 6.6.1, "Installing the DAC Platform and Oracle BI Applications Metadata Repository Files"](#)
- [Section 6.6.2, "Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica"](#)
- [Section 6.6.3, "Upgrading the DAC Repository"](#)

6.6.1 Installing the DAC Platform and Oracle BI Applications Metadata Repository Files

The current version of DAC is installed by its own installer and not the Oracle BI Applications installer. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

After you install DAC, you then need to copy metadata files from the machine hosting Oracle BI Applications to the machines hosting the DAC Client and Server. You then need to import the new metadata into the DAC Repository. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Installing and Setting Up the DAC Platform

Note: You must perform all of the procedures in this section, including installing JDBC drivers, creating ODBC database connections, and installing pmrep and pmcmd command line programs.

- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

6.6.2 Configuring the DAC Client and Server to Work With Oracle BI Applications and Informatica

You need to perform certain configuration tasks to enable the DAC Client and Server to work with Oracle BI Applications and Informatica PowerCenter. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring the SiebelUnicodeDB Custom Property

Note: This procedure is required only if your source to target data movement configuration is Unicode to Unicode.

- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

6.6.3 Upgrading the DAC Repository

You will use the Refresh Base option of the DAC's Upgrade/Merge Wizard to complete the upgrade of your existing DAC Repository. For information about how to use the Refresh Base option of the Upgrade/Merge Wizard, see the topic titled, "About the Refresh Base Option," in "Chapter 10: Upgrading, Comparing and Merging DAC Repositories," in the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

The *Oracle Business Intelligence Data Warehouse Administration Console User's Guide* is available in the Oracle Business Intelligence Data Warehouse Administration Console Documentation Library on the Oracle Technology Network.

Before you upgrade the DAC Repository, do the following:

- Make a copy of your default source system container. (You cannot change the metadata in the preconfigured container. You must make a copy of it in order to be able to modify the objects it contains.) For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

6.7 Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases

This section contains the following topics:

- [Section 6.7.1, "Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases"](#)
- [Section 6.7.2, "Upgrading the Data Warehouse Schema From Version 7.9.x to 7.9.3 and Migrating Data"](#)
- [Section 6.7.3, "Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data"](#)
- [Section 6.7.4, "Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data"](#)
- [Section 6.7.5, "Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data"](#)
- [Section 6.7.6, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data"](#)
- [Section 6.7.7, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data"](#) (Applicable only if you are using an Oracle database as your data warehouse database.)
- [Section 6.7.8, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data"](#)
- [Section 6.7.9, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data"](#)

If you are currently running Oracle BI Applications release 7.9.0, 7.9.1 or 7.9.2, you need to perform the procedures in [Section 6.7.1](#), [Section 6.7.2](#), [Section 6.7.3](#), [Section 6.7.4](#), [Section 6.7.5](#), [Section 6.7.6](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.3, you need to perform the procedures in [Section 6.7.1](#), [Section 6.7.3](#), [Section 6.7.4](#), [Section 6.7.5](#), [Section 6.7.6](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.4, you need to perform the procedures in [Section 6.7.1](#), [Section 6.7.4](#), [Section 6.7.5](#), and [Section 6.7.6](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.5, you need to perform the procedures in [Section 6.7.1](#) and [Section 6.7.5](#), and [Section 6.7.6](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.5.1, you need to perform the procedures in [Section 6.7.1](#) and [Section 6.7.6](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.6, you need to perform the procedures in [Section 6.7.1](#), [Section 6.7.7](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.6.1, you need to perform the procedures in [Section 6.7.1](#), [Section 6.7.8](#), and [Section 6.7.9](#).

If you are currently running Oracle BI Applications release 7.9.6.2, you need to perform the procedures in [Section 6.7.1](#) and [Section 6.7.9](#).

These procedures add new tables, columns, and indexes to the existing data warehouse schema. They also modify the existing data warehouse schema objects.

6.7.1 Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases

The steps in this procedure are required for upgrading to Oracle BI Applications release 7.9.6.3 from all previous releases.

To perform common data warehouse upgrade steps

1. If you are using a DB2 database, you need to execute the stored procedure siebproc.sql, which creates procedures and functions needed by the DDLimp utility. This file is located in \dwrep\siebproc\db2udb.
2. Copy all of the domain value files in the folder \OracleBI\dwrep\Informatica\LkpFiles into the folder <INFA_HOME>\server\infa_shared\LkpFiles.
3. Restore the Upgrade repository:
 - a. Copy the file Upgrade.rep file from the folder OracleBI\dwrep\Upgrade\Informatica\Repository into the folder <INFA_HOME>\server\infa_shared\Backup.
 - b. Launch the Informatica Administrator, and restore Upgrade.rep (located in <INFA_HOME>\server\infa_shared\Backup).
4. In Informatica Workflow Manager, open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and edit the connect string, user and password for the relational connections as follows:
 - a. For the OLTP connection:

If you are using the Oracle EBS 11i source system, edit the connection PARAM_OLTP_ORA11i.

If you are using Oracle EBS R12, source system, edit the connection PARAM_OLTP_ORAR12.
 - b. For the OLAP connection, edit the connection PARAM_OLAP.
 - c. For the DAC database connection, edit the connection PARAM_DAC.

Note: If you are connected to an Oracle database, use the Oracle native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

6.7.2 Upgrading the Data Warehouse Schema From Version 7.9.x to 7.9.3 and Migrating Data

Perform this procedure if you are upgrading from Oracle BI Applications releases 7.9.0, 7.9.1 or 7.9.2.

To upgrade the data warehouse schema and migrate data into the upgraded data warehouse

1. Run the UPGRADE_793.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the ddl_793.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
3. Run the 793_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
4. Migrate data into the upgraded data warehouse.
 - a. Configure the Informatica parameter file 793_UPG_PARAMS.txt. For instructions, see [Appendix A.1.1, "Setting Parameters in 793_UPG_PARAMS.txt."](#)
 - b. In Informatica Workflow Manager, navigate to the folder UPGRADE_790_to_793 and run the Update_Dimensions workflow, and then run the Update_Facts workflow.
 - c. If you are using the Oracle EBS 11.5.10 source system, navigate to the folder UPGRADE_790_to_793_ORA11510, and run the SIL_PurchaseAmount_Patch workflow.
5. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\SessLogs directory indicate errors or failures.
 - c. Check whether the script 793_UPGRADE_PRE_DIMENSION_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log file for the scripts ddl_793.ctf and UPGRADE_793.ctf, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

6. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W_xxxx_x_79x tables, LKP_xxxx_x and 79x_XXXX_TMP.

This step frees the space occupied by these backup tables.

6.7.3 Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data

Perform the procedure in this section to upgrade from Oracle BI Applications release 7.9.3 to 7.9.4.

To upgrade the data warehouse schema and migrate data into the upgraded data warehouse

1. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
2. Run the `794_UPGRADE_PRE_CTL_SCRIPT.sql`. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
3. Run the `UPGRADE_794.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the `ddl_794.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the `794_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
6. Migrate data into the upgraded data warehouse.
 - a. Configure the Informatica parameter file `794_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.2, "Setting Parameters in 794_UPG_PARAMS.txt."](#)
 - b. In Informatica Workflow Manager, navigate to the folder `UPGRADE_793_to_794_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
7. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the `<INFA_HOME>\server\infa_shared\SessLogs` directory indicate errors or failures.
 - c. Check whether the script `794_UPGRADE_PRE_DIMENSION_SCRIPT.sql` that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log file for the scripts `ddl_794` and `UpgradPGRAD_794.ctl`, which you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

8. If the data migration was successful, drop the tables that were created during the upgrade process, such as, `W_xxxx_x_79x` tables, `LKP_xxxx_x` and `79x_XXXX_TMP`. This step frees the space occupied by these backup tables.
9. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

6.7.4 Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5. and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.4 to 7.9.5.

To upgrade the data warehouse schema and migrate data into the upgraded data warehouse

1. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
2. Run the `795_UPGRADE_PRE_CTL_SCRIPT.sql`. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
3. Run the `UPGRADE_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the `ddl_795.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
6. Migrate data into the upgraded data warehouse.
 - a. Configure the Informatica parameter file `795_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
 - b. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_794_to_795_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_794_to_795_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
7. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the `<INFA_HOME>\server\infa_shared\Sesslogs` directory indicates errors or failures.
 - c. Check whether the script `795_UPGRADE_PRE_DIMENSION_SCRIPT.sql` that you ran in the SQL client of the database failed or errored out while executing.
 - d. Check the log file for the scripts `ddl_795.ctl` and `UPGRADE_795.ctl`, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

8. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

6.7.5 Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.5 to 7.9.5.1.

To upgrade the data warehouse schema and migrate data into the upgraded data warehouse

1. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
2. Run the 7951_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 3. Run the UPGRADE_7951.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
- 4. Run the ddl_7951.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
- 5. Run the 7951_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 6. Migrate data into the upgraded data warehouse:
 - a. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
 - b. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_7951_ORA11i and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_7951_ORAR12 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
- 7. Verify the data migrated successfully.
 - a. Validate the data in the upgraded data warehouse.
 - b. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
 - c. Check whether the script 7951_UPGRADE_PRE_DIMENSION_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.

- d. Check the log file for the scripts ddl_7951.ctl and Upgrade_7951.ctl, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

8. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.
This step frees the space occupied by these backup tables.
9. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

6.7.6 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.5.1 to 7.9.6.1.

This section includes the following procedures:

- [Section 6.7.6.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 6.7.6.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 6.7.6.3, "Verifying the Data Migrated Successfully"](#)
- [Section 6.7.6.4, "Resetting Refresh Dates"](#)

6.7.6.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

Note: Upgrading to version 7.9.6.1 may involve running scripts that have a "796_" prefix.

To upgrade the data warehouse schema to version 7.9.6.1

1. If you are using an Oracle database, run the 796_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
 - c. Open the 796_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. If you are using a SQL Server or DB2 database, run the 7961_UPGRADE_DROP_INDEXES.sql script.
 - a. Open the SQL client for your database type.

- b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
 - c. Open the 7961_UPGRADE_DROP_INDEXES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
3. If you are using a SQL Server database, run the 7961_UPGRADE_PRE_UPG_CTL_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
 4. Run the UPGRADE_796.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
 5. Run the 7961_UPGRADE_PRE_CTL_SCRIPT.sql:
 - a. Go to OracleBI\dwrep\Upgrade\DbScripts\<<database type> folder, and open the 7961_UPGRADE_PRE_CTL_SCRIPT.sql file.
 - b. Search for the Datasource_Num_ID parameter, and make sure the value is correct for your existing implementation.
 - c. If you are using a DB2 database, search for the text "TBS_32K," and replace it with the appropriate 32K table space name that you have configured.
 - d. Execute the script.
 6. Run the ddl_7961.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

6.7.6.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 796_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 796_UPG_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
4. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle EBS 11i Source Systems."](#)
5. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
6. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.3, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
7. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.2, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Project Analytics."](#)

8. Back up and truncate the table W_POSITION_DH.
9. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_ORA11i and execute the following workflows in the order they appear:
 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_ORAR12 and execute the following workflows in the order they appear:

 - UPGRADE_DIMENSIONS
 - UPGRADE_FACTS
10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
11. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
12. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
13. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
14. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
15. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Section A.3.3, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
16. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Project Analytics."](#)
17. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
18. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_ORA11i and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_ORAR12 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
19. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example,

Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)

20. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\

6.7.6.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961_UPGRADE_DROP_INDEXES.sql

7961_UPGRADE_PRE_CTL_SCRIPT.sql

796_UPGRADE_PRE_DIMENSION.sql

796_UPGRADE_POST_SCRIPT.sql

7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the ddl_7961.ctf and Upgrade_796.ctf scripts that you ran using the DDLimp command to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6.7.6.4 Resetting Refresh Dates

After verifying the data was migrated successfully into the upgraded data warehouse, follow this procedure to reset refresh dates.

1. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_7951_TO_796_ORA11i and execute the RESET_DAC_REFRESH_DATES workflow.
2. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_7951_TO_796_ORAR12 and execute the RESET_DAC_REFRESH_DATES workflow.
3. In DAC, do the following:

- a. Navigate to the Setup view, and click the Physical Data Sources tab.
 - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
 - c. Click the Refresh Dates subtab (in the lower pane).
 - d. Query for the table FND_LOOKUP_VALUES, and set the value in the Refresh Date column to NULL.
4. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

6.7.7 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6 to release 7.9.6.1.

Note: This procedure is applicable only if you are using an Oracle database for your data warehouse database.

This section includes the following procedures:

- [Section 6.7.7.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 6.7.7.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 6.7.7.3, "Verifying the Data Migrated Successfully"](#)

6.7.7.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

To upgrade the data warehouse schema to version 7.9.6.1

1. Run the UPGRADE_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
2. Run the 796ORA_TO_7961ORA_UPGRADE_PRE_CTL_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
3. Run the ddl_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

6.7.7.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7961_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7961_UPG_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
4. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
5. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Section A.3.3, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
6. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796_UPG_PARAMS.txt for Oracle Project Analytics."](#)
7. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_ORA11i and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_796_to_7961_ORAR12 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
9. Run the 7961_UPGRADE_POST_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

6.7.7.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
 - 7961_UPGRADE_DROP_INDEXES.sql
 - 7961_UPGRADE_PRE_CTL_SCRIPT.sql
 - 7961_UPGRADE_PRE_DIMENSION.sql
 - 7961_UPGRADE_POST_SCRIPT.sql

4. Check the log files for the `ddl_7961.ctl` and `Upgrade_7961.ctl` scripts, which you ran using the `DDLimp` command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as `W_xxxx_x_79x`, `LKP_xxxx_x` and `79x_xxxx_TMP`.

This step frees the space occupied by these backup tables.

6.7.8 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.1 to 7.9.6.2.

This section contains the following procedures:

- [Section 6.7.8.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 6.7.8.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 6.7.8.3, "Verifying the Data Migrated Successfully"](#)

6.7.8.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

To upgrade the data warehouse schema to version 7.9.6.2

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the `7962_UPGRADE_PRE_UPG_CTL_SCRIPT.sql` file. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\SQLServer`.
4. Run the `UPGRADE_7962.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the `7962_UPGRADE_PRE_CTL_SCRIPT.sql`. This file is located in the folder `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
6. Run the `DDL_7962.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

6.7.8.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7962_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
2. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7962_UPG_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
4. If you are upgrading Oracle Financial Analytics, do the following:
 - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
 - b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
5. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Appendix A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
6. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
7. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_ORA11i and execute the first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_ORAR12 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
8. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
9. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

6.7.8.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql

7962_UPGRADE_PRE_CTL_SCRIPT.sql

7962_UPGRADE_PRE_DIMENSION.sql

createETLIndexes.sql

createQueryIndexes.sql

4. Check the log files for the ddl_7962.ctf and Upgrade_7962.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

6.7.9 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Section 6.7.9.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.3"](#)
- [Section 6.7.9.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 6.7.9.3, "Verifying the Data Migrated Successfully"](#)

6.7.9.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

Note: You must drop the query indexes before running the scripts to upgrade the schema.

3. If you are using a SQL Server database, run the 7963_UPGRADE_PRE_UPG_CTL_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963_UPGRADE_PRE_CTL_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
6. Run the DDL_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

6.7.9.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

To migrate data into the upgraded data warehouse

1. Run the 7963_UPGRADE_PRE_DIMENSION_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
2. Run the 7963_UPGRADE_DROP_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<database type>.
3. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
4. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
5. If you are upgrading Oracle Financial Analytics, do the following:
 - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
 - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
6. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Human Resources Analytics."](#)
7. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)
8. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_ORA11i and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 If you are using Oracle EBS R12x, in Informatica Workflow Manager, navigate to the folder UPGRADE_7962_to_7963_ORAR12 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
9. Upgrade geography-related attributes if necessary for your environment:

- a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963_UPGRADE_POST_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963_UPGRADE_POST_SCRIPT.sql script.
10. Run the 7963_UPGRADE_CREATE_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 11. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep). For instructions, see [Appendix D.1, "Running reset_infa_seq_gen.bat on Base Informatica Repository."](#)
- 12. Run the createQueryIndices.sql script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

6.7.9.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA_HOME>\server\infa_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

```
7963_UPGRADE_PRE_CTL_SCRIPT.sql
7963_UPGRADE_PRE_DIMENSION.sql
7963_UPGRADE_DROP_ETLINDICES.sql
7963_UPGRADE_CREATE_ETLINDICES.sql
dropQueryIndices.sql
createQueryIndices.sql
```

4. Check the log files for the ddl_7963.ctl and Upgrade_7963.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

Note: Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W_xxxx_x_79x, LKP_xxxx_x and 79x_xxxx_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
 - a. In the Informatica Administrator, select the Upgrade repository service.
 - b. In the General Properties area of the Properties tab, click Edit.
 - c. Make sure the operating mode of the repository service is set to Exclusive.
 - d. Click OK.
 - e. Choose Actions, and then click Delete Contents.
 - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

6.8 Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases

If your data warehouse database platform is Teradata, you can upgrade to Oracle BI Applications version 7.9.6.1 from Oracle BI Applications 7.9.x.

This section includes the following topics:

- [Section 6.8.1, "Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3"](#)
- [Section 6.8.2, "Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3"](#)
- [Section 6.8.3, "Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3"](#)
- [Section 6.8.4, "Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3"](#)
- [Section 6.8.5, "Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3"](#)
- [Section 6.8.6, "Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3"](#)
- [Section 6.8.7, "Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3"](#)

6.8.1 Upgrading Oracle BI Applications From Version 7.9.0 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.0 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
 - 793_UPGRADE_TMP_TABLES.sql
 - 794_UPGRADE_TMP_TABLES.sql
 - 795_UPGRADE_TMP_TABLES.sql

- 7951_UPGRADE_TMP_TABLES.sql
 - 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 793_RENAME_AND_EXCEPTIONS.sql script.
- a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 793_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.
- ```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
- d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 793\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
- a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
- The Configuration tab of the Data Warehouse Configuration Wizard is active.
- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
- The Data Warehouse SQL tab is active.
- e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
    - 794\_UPDATE\_DEFAULTS.sql
    - 795\_UPDATE\_DEFAULTS.sql
    - 7951\_UPDATE\_DEFAULTS.sql
    - 7961\_UPDATE\_DEFAULTS.sql
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  6. Execute the 793\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)

8. Configure the Informatica parameter file 793\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.1, "Setting Parameters in 793\\_UPG\\_PARAMS.txt."](#)
9. In Informatica Workflow Manager, navigate to the folder UPGRADE\_790\_to\_793 and run the Update\_Dimensions workflow, and then run the Update\_Facts workflow.
10. If you are using the Oracle EBS 11.5.10 source system, navigate to the folder UPGRADE\_790\_to\_793\_ORA11510, and run the SIL\_PurchaseAmount\_Patch workflow.
11. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the 794\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
13. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
14. Configure the Informatica parameter file 794\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.2, "Setting Parameters in 794\\_UPG\\_PARAMS.txt."](#)
15. In Informatica Workflow Manager, navigate to the folder UPGRADE\_793\_to\_794\_ORA11i and execute the UPGRADE\_DIMENSIONS and UPGRADE\_FACTS workflows.
16. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the 795\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata
18. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Configure the Informatica parameter file 795\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt."](#)
20. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
21. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
22. Execute the 7951\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
23. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
24. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
25. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
26. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

27. Execute the 796\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
28. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
30. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems."](#)
31. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
32. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
33. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
34. Back up and truncate the table W\_POSITION\_DH.
35. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORA11i and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
36. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORAR12 and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
37. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
38. Execute the 7961\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
39. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade.
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
40. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
41. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
42. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
43. Execute the 7962\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
44. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
45. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
46. If you are upgrading Oracle Financial Analytics, do the following:
- a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)

- b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
- 47. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
- 48. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 49. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 50. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 51. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 52. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
- 53. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
- 54. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
- 55. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)



56. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

57. Upgrade geography-related attributes if necessary for your environment:
- a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.

58. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

59. In DAC, do the following:
- a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the FND\_LOOKUP\_VALUES table, and set the value in the Refresh Date column to NULL.
60. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
61. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.

- d. Click OK.
- e. Choose Actions, and then click Delete Contents.
- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 6.8.2 Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.3 to version 7.9.6.3 if your data warehouse database platform is Teradata.

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 794\_UPGRADE\_TMP\_TABLES.sql
  - 795\_UPGRADE\_TMP\_TABLES.sql
  - 7951\_UPGRADE\_TMP\_TABLES.sql
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 794\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 794\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 794\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
  - a. Log into DAC.

- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
- i. Execute the scripts in the upgrade-regular.sql file.
- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
- k. Recreate STATISTICS on any column on which they were dropped.

- 5. Execute the following scripts:

- 794\_UPDATE\_DEFAULTS.sql
- 795\_UPDATE\_DEFAULTS.sql
- 7951\_UPDATE\_DEFAULTS.sql

- 7961\_UPDATE\_DEFAULTS.sql
  - 7962\_UPDATE\_DEFAULTS.sql
  - 7963\_UPDATE\_DEFAULTS.sql
6. Execute the 794\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  8. Configure the Informatica parameter file 794\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.2, "Setting Parameters in 794\\_UPG\\_PARAMS.txt."](#)
  9. In Informatica Workflow Manager, navigate to the folder UPGRADE\_793\_to\_794\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
  11. Execute the 795\_Backup Tables.sql script.
  12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  13. Configure the Informatica parameter file 795\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt."](#)
  14. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
 If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

15. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
16. Execute the `7951_Backup Tables.sql` script.
17. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
18. Configure the Informatica parameter file `7951_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
19. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_795_to_7951_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_795_to_7951_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
20. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
21. Execute the `796_Backup Tables.sql` script.
22. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
23. Configure the Informatica parameter file `796_UPG_PARAMS.txt` file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)

24. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems."](#)
25. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
26. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
27. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
28. Back up and truncate the table W\_POSITION\_DH.
29. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORA11i and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
30. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORAR12 and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
31. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
32. Execute the 7961\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
33. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)

34. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
35. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
36. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
37. Execute the 7962\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
38. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository.
39. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
40. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
  - b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
41. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
42. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
43. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 44. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 45. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 46. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
- 47. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
- 48. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
- 49. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)
- 50. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 51. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.



52. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
53. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the `FND_LOOKUP_VALUES` table, and set the value in the Refresh Date column to NULL.
54. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

### 6.8.3 Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.4 to version 7.9.6.3 if your data warehouse database platform is Teradata.

#### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - `795_UPGRADE_TMP_TABLES.sql`
  - `7951_UPGRADE_TMP_TABLES.sql`
  - `7961_UPGRADE_TMP_TABLES.sql`
  - `7962_UPGRADE_TMP_TABLES.sql`
  - `7963_UPGRADE_TMP_TABLES.sql`
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
  - c. Open the `<version>_UPGRADE_TMP_TABLES.sql` file, and copy the contents into the SQL client.
  - d. Execute the script.

2. Edit and execute the 795\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 795\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 795\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
 

The Configuration tab of the Data Warehouse Configuration Wizard is active.
  - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
 

The Data Warehouse SQL tab is active.
  - e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
    - 795\_UPDATE\_DEFAULTS.sql
    - 7951\_UPDATE\_DEFAULTS.sql
    - 7961\_UPDATE\_DEFAULTS.sql
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  6. Execute the 795\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  8. Configure the Informatica parameter file 795\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt."](#)
  9. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
 

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
11. Execute the 7951\_Backup Tables.sql script.
  12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  13. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
  14. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  15. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
  16. Execute the 796\_Backup Tables.sql script.
  17. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  18. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
  19. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems."](#)

20. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
21. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Section A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
22. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
23. Back up and truncate the table W\_POSITION\_DH.
24. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORA11i and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
25. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORAR12 and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
26. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
27. Execute the 7961\_Backup Tables.sql script.
28. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)

30. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
31. Execute the `7962_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
32. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
33. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
34. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
  - b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
35. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
36. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
37. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
38. Execute the `7963_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.

39. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
40. Configure the Informatica parameter file `7963_UPG_PARAMS.txt`. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
41. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
42. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
43. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)
44. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
45. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the `7963_UPGRADE_POST_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.  
  
If the query returns a count of zero, you do not need to run the `7963_UPGRADE_POST_SCRIPT.sql` script.
46. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
47. In DAC, do the following:
- a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the table FND\_LOOKUP\_VALUES, and set the value in the Refresh Date column to NULL.
48. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

### 6.8.4 Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.5 to version 7.9.6.3 if your data warehouse database platform is Teradata.

#### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 7951\_UPGRADE\_TMP\_TABLES.sql
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.



- c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
  4. Generate upgrade scripts from DAC.
    - a. Log into DAC.
    - b. From the menu bar, select Tools, then ETL Management, and then Configure.
    - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
 

The Configuration tab of the Data Warehouse Configuration Wizard is active.
    - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
 

The Data Warehouse SQL tab is active.
    - e. Do the following:
      - Select the appropriate physical data source name for the Teradata database.
      - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
    - 7951\_UPDATE\_DEFAULTS.sql
    - 7961\_UPDATE\_DEFAULTS.sql
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  6. Execute the 7951\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  8. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
  9. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
 

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

11. Execute the 796\_Backup Tables.sql script.
12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
13. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
14. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems."](#)
15. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
16. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
17. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
18. Back up and truncate the table W\_POSITION\_DH.
19. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORA11i and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
20. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORAR12 and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
21. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 22. Execute the 7961\_Backup Tables.sql script.
- 23. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 24. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
- 25. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
 If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 26. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 27. Execute the 7962\_Backup Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 28. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 29. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt.
- 30. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)

- b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
31. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
32. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
33. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
34. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
35. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
36. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
37. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
38. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
39. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)

40. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORAR12 and execute execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

41. Upgrade geography-related attributes if necessary for your environment:
- a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.

42. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

43. In DAC, do the following:
- a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the table FND\_LOOKUP\_VALUES, and set the value in the Refresh Date column to NULL.
44. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 6.8.5 Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.5.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
 

The Configuration tab of the Data Warehouse Configuration Wizard is active.
  - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
 

The Data Warehouse SQL tab is active.

e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
- i. Execute the scripts in the upgrade-regular.sql file.
- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
- k. Recreate STATISTICS on any column on which they were dropped.

5. Execute the following scripts:

- 7961\_UPDATE\_DEFAULTS.sql
- 7962\_UPDATE\_DEFAULTS.sql
- 7963\_UPDATE\_DEFAULTS.sql

6. Execute the 796\_Backup Tables.sql script.

7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)



- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
8. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt file. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
9. If you are using Oracle EBS 11i, configure parameters specific to Oracle 11i source systems. For instructions, see [Appendix A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems."](#)
10. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Appendix A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
11. If you are upgrading Oracle Supply Chain and Order Management Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics."](#)
12. If you are deploying Oracle Project Analytics, you need to configure specific parameters. For more information, see [Appendix A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
13. Back up and truncate the table W\_POSITION\_DH.
14. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORA11i and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
15. If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_ORAR12 and execute the following workflows in the order specified below:
  - UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
16. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
17. Execute the 7961\_Backup Tables.sql script.
18. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 19. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
- 20. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 21. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 22. Execute the 7962\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
- 23. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 24. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
- 25. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
  - b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
- 26. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)

27. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_ORAR12` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

28. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the `7963_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
30. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file `7963_UPG_PARAMS.txt`. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
32. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
33. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
34. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)
35. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_ORA11i` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_ORAR12` and execute

first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

36. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<>database type>.
 

If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
37. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
38. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the table FND\_LOOKUP\_VALUES, and set the value in the Refresh Date column to NULL.
39. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 6.8.6 Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

**To upgrade the data warehouse schema and migrate data for a Teradata database**

1. Execute the 7962\_UPGRADE\_TMP\_TABLES.sql and 7963\_UPGRADE\_TMP\_TABLES.sql scripts.
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.  
The Configuration tab of the Data Warehouse Configuration Wizard is active.
  - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.  
The Data Warehouse SQL tab is active.
  - e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the 7962\_UPDATE\_DEFAULTS.sql and 7963\_UPDATE\_DEFAULTS.sql scripts.
  5. Execute the 7962\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  7. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
  8. If you are upgrading Oracle Financial Analytics, do the following:
    - a. Configure application-specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
    - b. Make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
  9. If you are using Oracle EBS R12, you need to configure specific parameters and update mappings for COGS. For instructions, see [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12."](#)
  10. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
 

If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  11. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  13. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  14. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
  15. If you are upgrading Oracle Financial Analytics, do the following:
    - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
    - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
  16. If you are upgrading Oracle Human Resources Analytics, set the parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
  17. If you are upgrading Oracle Procurement and Spend Analytics, set the parameters as specified in [Appendix A.3.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems."](#)
  18. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
 If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  19. Upgrade geography-related attributes if necessary for your environment:
    - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
- 20. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 21. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the table FND\_LOOKUP\_VALUES, and set the value in the Refresh Date column to NULL.
- 22. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

### 6.8.7 Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.2 to version 7.9.6.3 if your data warehouse database platform is Teradata.

#### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the 7963\_UPGRADE\_TMP\_TABLES.sql script.
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the 7963\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.



2. The `teradata.missing.defaults.properties` file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the `teradata.missing.defaults.properties` file to match your environment. The file is located in `\bifoundation\dac\conf\sqlgen\sql\teradata`.

3. Generate upgrade scripts from DAC.

a. Log into DAC.

b. From the menu bar, select Tools, then ETL Management, and then Configure.

c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in `\bifoundation\DAC\conf\sqlgen\sql\teradata`:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the `upgrade-regular.sql` file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the `upgrade-questionable.sql` file. It is for information purposes only.

f. Open the `upgrade-regular.sql` and `upgrade-questionable.sql` files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.

i. Execute the scripts in the `upgrade-regular.sql` file.

- j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the 7963\_UPDATE\_DEFAULTS.sql scripts.
  5. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  7. If you are upgrading Oracle Financial Analytics, do the following:
    - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
    - b. If you are using Oracle EBS 11.5.10 family pack OIE.I or OIE.J, set the parameters as specified in [Appendix A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J."](#)
  8. If you are using Oracle EBS 11i, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORA11i and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.  
 If you are using Oracle EBS R12, R12.1.1, or R12.1.2, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_ORAR12 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
  9. Upgrade geography-related attributes if necessary for your environment:
    - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
    - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
 

If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
  10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 11. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 6.9 Localizing the Policy Store

The Oracle BI Applications 7.9.6.3 installer contains different policy stores for different languages. The default policy store that is deployed during the installation is in English. If you want to configure the policy store for any of the other available languages, perform the procedure below.

### To configure the policy store for languages other than English

1. Stop the Oracle BI Services.
2. Copy the system-jazn-data\_<LN>.xml file from \$ORACLE\_BI\_HOME\biapps\admin\provisioning\localization\ to \$DOMAIN\_HOME\config\fmwconfig\.
 

For example, to implement the policy store for French, you would copy the file system-jazn-data\_fr.xml into the \fmwconfig directory.
3. Back up the existing system-jazn-data.xml file in \$DOMAIN\_HOME\config\fmwconfig\.
4. Rename \$DOMAIN\_HOME\config\fmwconfig\system-jazn-data\_<LN>.xml to system-jazn-data.xml.
5. Start Oracle BI Services.

## 6.10 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server embedded LDAP server.

### To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, myrealm).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).

4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

## 6.11 Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and web catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Oracle BI Repository. This repository is the one that you originally received from Oracle as part of your current Oracle BI Applications installation (the version from which you are upgrading). Standard repositories from previous releases are available in the installation folder \$ORACLE\_BI\_HOME\biapps\upgrade.
- The Oracle BI Repository you are currently using (the repository deployed in production).
- The Presentation Catalog you are currently using (the catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

---

---

**Note:** Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 6.9, "Localizing the Policy Store."](#)

---

---

## 6.12 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

---

---

**Note:** Before you begin this procedure, you must have already upgraded the Oracle BI repository you are currently using and the standard (out-of-the-box) Oracle BI repository version 7.9.x (the version from which you are upgrading) to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

---

This process merges your customizations of a prior release of the Oracle BI repository with the new release of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository.

To upgrade the repository, perform the following tasks:

- [Section 6.12.1, "Preparing for the Oracle BI Repository Upgrade"](#)
- [Section 6.12.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 6.12.3, "Comparing the Oracle BI Repositories"](#)
- [Section 6.12.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 6–1](#) provides the names and descriptions of the repositories used in the examples in this section.

**Table 6–1 Names of Analytics Repositories used in Examples**

| Name of Repository                 | Description                                                                                                                                                                                             |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OracleBIAnalyticsApps_79x.rpd      | The standard Oracle BI repository for the version you are upgrading from.<br><br><b>Note:</b> Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade. |
| OracleBIAnalyticsApps.rpd          | The standard Oracle BI repository for the version you are upgrading to.                                                                                                                                 |
| Customer_OracleBIAnalyticsApps.rpd | The Oracle BI repository that contains your customizations for the version you are upgrading from.                                                                                                      |
| Merged_Repository_OracleBI.rpd     | The Oracle BI repository that contains your customizations for the version you are upgrading to.                                                                                                        |

## 6.12.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

### To prepare for the Analytics repository upgrade

1. Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
  - Original
  - AfterEqualize
  - AfterMerge
  - AfterManualWork
  - AfterRegressions
2. Copy the following repositories to the folder \OracleBIUpgrade\Original:
  - The standard repository that you upgraded to the 11g format in [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#) (for example, OracleBIAnalyticsApps\_79x.rpd).
  - The production repository that you also upgraded to the 11g format in [Section 6.11](#) (for example, Customer\_OracleBIAnalyticsApps.rpd).

- The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Oracle BI Applications for one or more modules using a Oracle BI repository in which you extracted the corresponding projects for the modules from the standard Oracle BI repository file you received from the previous release, you need to extract the same projects from the OracleBIAnalyticsApps\_79x.rpd file and use this as your original repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Oracle BI repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the OracleBIAnalyticsApps\_79x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

## 6.12.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories (for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer\_OracleBIAnalyticsApps.rpd whether it is coming from the OracleBIAnalyticsApps\_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W\_DAY\_D" might be considered equivalent to "Core"."W DAY D" if the characters "\_" and " " have been declared as equivalent.
- **Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the equalizerpds executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

[Table 6–2](#) provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

**Table 6–2 Rename MAP Files to Be Used for Various Releases**

| <b>Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)</b> | <b>Rename MAP File to Be Used</b> |
|-----------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Siebel Business Analytics Applications 7.0.x                                                                    | Not available                     |
| Siebel Business Analytics Applications 7.5.x                                                                    | Not available                     |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)                                   | Rename77-7963.map                 |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)                                       | Rename771-7963.map                |
| Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release                         | Rename782-7963.map                |
| Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release                          | Rename783-7963.map                |
| Oracle BI Applications 7.9.0                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.1                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.2                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.3                                                                                    | Rename793to7963.map               |
| Oracle BI Applications 7.9.4                                                                                    | Rename794to7963.map               |
| Oracle BI Applications 7.9.5                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.5.1                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.5.2                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.6                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.6.2                                                                                  | Not required                      |

The syntax of the equalizerpds command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udml_utf8_file_name_equalization] [-O output_repository_name]
[-Y equalStringSet]
-Y Treat the characters as equals.
/? Display this usage information and exit.
```

### To equalize a repository

1. Copy the appropriate MAP file from the \$ORACLE\_BI\_HOME\biapps\upgrade folder into the folder where you will execute equalizerpds.exe, for example, \OracleBIUpgrade\Original.
2. Run equalizerpds.exe to equalize the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsAppss\_793.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder.

If the equalizerpds.exe executable file runs correctly, no errors are returned.

3. Run equalizerpds.exe to equalize your customized repository (for example, Customer\_OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsApps\_79x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_OracleBIAnalyticsApps.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_OracleBIAnalyticsApps.rpd
```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.

Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

### 6.12.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

#### To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

### 6.12.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

#### To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, OracleBIAnalyticsApps\_79x.rpd).



6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.
9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, `Merged_Repository_OracleBIAnalyticsApps.rpd`.

The new repository will contain the final results of the merge.

10. Select the **Equalize during merge** check box.
11. Click **Next**.
12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.

When all rows have a value in the Decision field, the Finish button is enabled.

14. Select the **Check consistency of the merged RPD** check box.
15. Click **Finish**.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

17. Copy the repository to the folder `\OracleBIUpgrade\AfterManualWork`.

## 6.13 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

---

You need to perform the following tasks to complete the presentation merge process.

- [Section 6.13.1, "Trimming the Input Presentation Catalog"](#)
- [Section 6.13.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 6.13.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This catalog is the output from the step you performed in [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the version of the unmodified presentation catalog you received with the Oracle BI Applications release you are upgrading from that you upgraded to the 11g format.
- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This catalog is the output from the step you performed in [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the presentation catalog currently in use at your organization that you upgraded to the 11g format.

---

---

**Caution:** In releases of Oracle BI Applications previous to 7.9, the Presentation Catalog (formerly known as the Siebel Analytics Web Catalog) was stored in a single file rather than in a directory structure of individual files. If you have a previous version of the Presentation Catalog, you will need to convert it to the new format. For more information about how to convert the Presentation Catalog to the new format, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*

---

---

### 6.13.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

## 6.13.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 6.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

### To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.

For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.
3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name>
-oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group
-new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

**Note:** If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

### 6.13.3 Testing the Results of the Presentation Catalog Upgrade

---

---

**Note:** Before you perform this step, you must first migrate the data into the upgraded data warehouse.

---

---

The Presentation Catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the Presentation Catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged Presentation Catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new Presentation Catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

## 6.14 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

#### To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the Presentation Catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE\_BI\_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.

5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAnalyticsApps.rpd and the production repository.



# Part III

---

## Upgrading When Your Source System is PeopleSoft

Part III contains instructions for upgrading to the current release of Oracle BI Applications when you are running PeopleSoft Enterprise Applications as your source system.

Part III contains [Chapter 7, "Upgrading Oracle BI Applications for PeopleSoft Source Systems."](#)

---

**Note:** Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at

<http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

---





---

---

# Upgrading Oracle BI Applications for PeopleSoft Source Systems

---

---

**Note:** Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

---

---

This section contains instructions for upgrading Oracle BI Applications when you are running PeopleSoft Enterprise Applications as your source system.

This section includes the following topics:

- [Section 7.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 7.2, "Upgrading Oracle BI Applications"](#)
- [Section 7.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 7.4, "Upgrading the Informatica Repository"](#)
- [Section 7.5, "Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC"](#)
- [Section 7.6, "Upgrading and Configuring DAC"](#)
- [Section 7.7, "Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases"](#)
- [Section 7.8, "Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases"](#)
- [Section 7.9, "Localizing the Policy Store"](#)
- [Section 7.10, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"](#)

- [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#)
- [Section 7.12, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"](#)
- [Section 7.13, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 7.14, "Regression Testing the Oracle BI Repository Merge"](#)

## 7.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

## 7.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release. This will be the backup of your DAC Client and Server and the DAC metadata files.)

## 7.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ significantly from previous versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

### To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 7.4, "Upgrading the Informatica Repository."](#)

In [Section 7.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle\_BI\_DW\_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

## 7.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

### To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

---



---

**Note:** This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 7.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle\_BI\_DW\_Base repository that you restore in the steps below.

---



---

2. Copy the Oracle\_BI\_DW\_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA\_HOME>\server\infa\_shared\Backup.

---

**Note:** The Oracle\_BI\_DW\_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 7.2, "Upgrading Oracle BI Applications."](#)

---

3. Restore the Oracle\_BI\_DW\_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
  - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
  - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.
  - c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

## 7.5 Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

## 7.6 Upgrading and Configuring DAC

This section includes instructions you must follow to upgrade and configure DAC. It includes the following topics:

- [Section 7.6.1, "Installing the DAC Platform and Oracle BI Applications Metadata Repository Files"](#)
- [Section 7.6.2, "Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica"](#)
- [Section 7.6.3, "Upgrading the DAC Repository"](#)

### 7.6.1 Installing the DAC Platform and Oracle BI Applications Metadata Repository Files

The current version of DAC is installed by its own installer and not the Oracle BI Applications installer. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

After you install DAC, you then need to copy metadata files from the machine hosting Oracle BI Applications to the machines hosting the DAC Client and Server. You then need to import the new metadata into the DAC Repository. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Installing and Setting Up the DAC Platform

---



---

**Note:** You must perform all of the procedures in this section, including installing JDBC drivers, creating ODBC database connections, and installing pmrep and pmcmd command line programs.

---



---

- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

## 7.6.2 Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica

You need to perform certain configuration tasks to enable the DAC Client and Server to work with Oracle BI Applications and Informatica PowerCenter. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring the SiebelUnicodeDB Custom Property

---



---

**Note:** This procedure is required only if your source to target data movement configuration is Unicode to Unicode.

---



---

- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

## 7.6.3 Upgrading the DAC Repository

You will use the Refresh Base option of the DAC's Upgrade/Merge Wizard to complete the upgrade of your existing DAC Repository. For information about how to use the Refresh Base option of the Upgrade/Merge Wizard, see the topic titled, "About the Refresh Base Option," in "Chapter 10: Upgrading, Comparing and Merging DAC Repositories," in the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

The *Oracle Business Intelligence Data Warehouse Administration Console User's Guide* is available in the Oracle Business Intelligence Data Warehouse Administration Console Documentation Library on the Oracle Technology Network.

Before you upgrade the DAC Repository, do the following:

- Make a copy of your default source system container. (You cannot change the metadata in the preconfigured container. You must make a copy of it in order to be able to modify the objects it contains.) For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

## 7.7 Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases

This section contains the following topics:

- [Section 7.7.1, "Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases"](#)
- [Section 7.7.2, "Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data"](#)
- [Section 7.7.3, "Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data"](#)
- [Section 7.7.4, "Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data"](#)
- [Section 7.7.5, "Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data"](#)
- [Section 7.7.6, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data"](#) (Applicable only if you are using an Oracle database as your data warehouse database.)
- [Section 7.7.7, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data"](#)
- [Section 7.7.8, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data"](#)

If you are currently running Oracle BI Applications version 7.9.3, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.2](#), [Section 7.7.3](#), [Section 7.7.4](#), [Section 7.7.5](#), [Section 7.7.7](#), and [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.4, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.3](#), [Section 7.7.4](#) and [Section 7.7.5](#), [Section 7.7.7](#), and [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.5, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.4](#) and [Section 7.7.5](#), [Section 7.7.7](#), and [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.5.1, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.5](#), [Section 7.7.7](#), [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.6, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.6](#), [Section 7.7.7](#), and [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.6.1, you need to perform the procedures in [Section 7.7.1](#), [Section 7.7.7](#), and [Section 7.7.8](#).

If you are currently running Oracle BI Applications version 7.9.6.2, you need to perform the procedures in [Section 7.7.1](#) and [Section 7.7.8](#).

These procedures add new tables, columns, and indexes to the existing data warehouse schema. They also modify the existing data warehouse schema objects.

## 7.7.1 Performing Data Warehouse Schema Upgrade Steps Common to All Previous Releases

The steps in this procedure are required for upgrading to Oracle BI Applications release 7.9.6.3 from all previous releases.

### To perform common data warehouse upgrade steps

1. Copy all of the domain value files in the folder `\OracleBI\dwrep\Informatica\LkpFiles` into the folder `<INFA_HOME>\server\infa_shared\LkpFiles`.
2. Restore the Upgrade repository:
  - a. Copy the file `Upgrade.rep` file from the folder `OracleBI\dwrep\Upgrade\Informatica\Repository` into the folder `<INFA_HOME>\server\infa_shared\Backup`.
  - b. Launch the Informatica Administrator, and restore `Upgrade.rep` (located in `<INFA_HOME>\server\infa_shared\Backup`).
3. Configure connections for the OLAP and OLTP databases.

---

**Note:** Make sure all the PeopleSoft source (OLTP) connections are created as Application connections and the OLAP connections are created as Relational connections.

---

- a. In Informatica Workflow Manager, open the Application Connection Browser (in the menu bar, select Connections, and then select Application), and configure the parameter `PARAM_OLTP` to match your OLTP environment.
- b. Open the Relational Connection Browser (in the menu bar, select Connections, and then select Relational), and configure the parameter `PARAM_OLAP` to match your OLAP environment.
- c. If you are upgrading Oracle Financial Analytics, configure `PARAM_OLTP_PFSF_FIN` to point to the PeopleSoft Financials Pillar database.
 

Note: This connection is present as both an Application connection and Relational connection. You must configure both of them to point to the PeopleSoft Financial Pillar database.
- d. If you are upgrading Oracle Human Resources Analytics, configure `PARAM_OLTP_PSFT_HR` to point to the PeopleSoft HRMS Pillar database.
 

Note: This connection is present as an Application connection only.
- e. Configure the Relational connection `PARAM_DAC` to match your DAC database connection.
- f. Configure a Relational connection with the same name defined in the DAC Physical Data Sources tab for the OLAP connection. The Type specified in DAC is "Warehouse." The default name for this connection is "Datawarehouse."
- g. Configure an Application connection with the same name defined in the DAC Physical Data Sources tab for the OLTP connection. The Type specified in DAC is "Source."

---



---

**Note:** If you are connected to an Oracle database, use the Oracle Native driver instead of ODBC.

If you are connected to a SQL Server database, use the ODBC driver rather than the native SQL Server driver.

---



---

4. Make sure all the date parameters are set up with an appropriate format.

For example, the format for \$\$LOW\_DATE is as follows:

```
SIL_EmployeeDimension_Full]
$$TYPE2_FLG=Y
$DBConnection_OLAP=adevuni_f11
MPLT_GET_ETL_PROC_WID.$$ETL_PROC_WID=1
$$LOW_DATE=to_date('01/01/1970','MM/DD/YYYY')
mplt_SIL_EmployeeDimension.$$LANGUAGE_CODE=E
mplt_SIL_EmployeeDimension.$$MASTER_CODE_NOT_FOUND=NA
mplt_SIL_EmployeeDimension.$$SOURCE_CODE_NOT_SUPPLIED=NA
```

## 7.7.2 Upgrading the Data Warehouse Schema From Version 7.9.3 to 7.9.4 and Migrating Data

Follow this procedure to upgrade from Oracle BI Applications release 7.9.3 to 7.9.4.

### To upgrade the data warehouse schema and migrate data into the upgraded data warehouse

1. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
2. Run the 794\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
3. Run the UPGRADE\_794.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run ddl\_794.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 794\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in the OracleBI\dwrep\Upgrade\DbScripts\<<database type> folder.
6. Migrate data into the upgraded data warehouse.
  - a. Configure the Informatica parameter file 794\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.2, "Setting Parameters in 794\\_UPG\\_PARAMS.txt."](#)
  - b. In Informatica Workflow Manager, navigate to the folder UPGRADE\_793\_to\_794\_PSFT88, and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
7. Verify the data migrated successfully.
  - a. Validate the data in the upgraded data warehouse.
  - b. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\SessLogs directory indicate errors or failures.



- c. Check whether the script 794\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
- d. Check the log file for the ddl\_794.ctf and Upgrade\_794.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

8. If the data migration was successful, drop the tables that were created during the upgrade process, such as, W\_xxxx\_x\_79x tables, LKP\_xxxx\_x and 79x\_XXXX\_TMP. This step frees the space occupied by these backup tables.
9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)

### 7.7.3 Upgrading the Data Warehouse Schema From Version 7.9.4 to 7.9.5 and Migrating Data

Follow this procedure to upgrade from Oracle BI Applications release 7.9.4 to 7.9.5.

**To upgrade the data warehouse schema and migrate data into the upgraded data warehouse**

1. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
2. Run the 795\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the OracleBI\dwrep\Upgrade\DbScripts\<<database type> folder.
3. Run the UPGRADE\_795.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the ddl\_795.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 795\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
6. Migrate data into the upgraded data warehouse.
  - a. Configure the Informatica parameter file 795\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt."](#)
  - b. In Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
7. Verify the data migrated successfully.

- a. Validate the data in the upgraded data warehouse.
  - b. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
  - c. Check whether the script 795\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
  - d. Check the log file for the ddl\_795.ctl and Upgrade\_795.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.
- If you did not detect any errors or failures in the steps above, then the data migration was successful.
8. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.  
This step frees the space occupied by these backup tables.
  9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)

## 7.7.4 Upgrading the Data Warehouse Schema From Version 7.9.5 to 7.9.5.1 and Migrating Data

Follow this procedure to upgrade from Oracle BI Applications release 7.9.5 to 7.9.5.1.

### To upgrade the data warehouse schema and migrate data

1. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
2. Run the 7951\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
3. Run the UPGRADE\_7951.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
4. Run the ddl\_7951.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7951\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
6. Migrate data into the upgraded data warehouse.
  - a. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
  - b. In Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_PSFT and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
7. Verify the data migrated successfully.
  - a. Validate the data in the upgraded data warehouse.
  - b. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.

- c. Check whether the script 7951\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql that you ran in the SQL client of the database failed or errored out while executing.
- d. Check the log file for the ddl\_7951.ctf and Upgrade\_7951.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

8. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.  
This step frees the space occupied by these backup tables.
9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)

## 7.7.5 Upgrading the Data Warehouse Schema From Version 7.9.5.1 to 7.9.6.1 and Migrating Data

Follow this procedure to upgrade from Oracle BI Applications release 7.9.5.1 to 7.9.6.1.

This section includes the following topics:

- [Section 7.7.5.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 7.7.5.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 7.7.5.3, "Verifying the Data Migrated Successfully"](#)

### 7.7.5.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

**Note:** Upgrading to version 7.9.6.1 may involve running scripts that have a "796\_" prefix.

#### To upgrade the data warehouse schema to version 7.9.6.1

1. If you are using an Oracle database, run the 796\_UPGRADE\_DROP\_INDEXES.sql script.
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
  - c. Open the 796\_UPGRADE\_DROP\_INDEXES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. If you are using a SQL Server or DB2 database, run the 7961\_UPGRADE\_DROP\_INDEXES.sql script.

- a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
  - c. Open the 7961\_UPGRADE\_DROP\_INDEXES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
3. If you are using a SQL Server database, run the 7961\_UPGRADE\_PRE\_UPG\_CTL\_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
  4. Run the UPGRADE\_796.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
  5. Run the 7961\_UPGRADE\_PRE\_CTL\_SCRIPT.sql:
    - a. Go to OracleBI\dwrep\Upgrade\DbScripts\<<database type> folder, and open the 7961\_UPGRADE\_PRE\_CTL\_SCRIPT.sql file.
    - b. Search for the Datasource\_Num\_ID parameter, and make sure the value is correct for your existing implementation.
    - c. If you are using a DB2 database, search for the text "TBS\_32K," and replace it with the appropriate 32K table space name that you have configured.
    - d. Execute the script.
  6. Run the ddl\_7961.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
  7. Run the 796\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<<database type>.

### 7.7.5.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
2. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
3. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
4. If you are upgrading Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
5. Back up and truncate table W\_POSITION\_DH.
6. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT88 and execute the following workflows in the order as they appear:
  - UPGRADE\_DIMENSIONS

- UPGRADE\_FACTS
- RESET\_DAC\_REFRESH\_DATES

If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT90 and execute the following workflows in the order as they appear:

- UPGRADE\_DIMENSIONS
  - UPGRADE\_FACTS
  - RESET\_DAC\_REFRESH\_DATES
7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
  8. Run the 7961\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.
  9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
  10. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
  11. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT88 and execute the first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft versions 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
  12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
  13. Run the 7961\_UPGRADE\_POST\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\<database type>.

### 7.7.5.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961\_UPGRADE\_DROP\_INDEXES.sql

7961\_UPGRADE\_PRE\_CTL\_SCRIPT.sql

7961\_UPGRADE\_PRE\_DIMENSION.sql

7961\_UPGRADE\_POST\_SCRIPT.sql

4. Check the log files for the ddl\_7961.ctl and Upgrade\_7961.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---



---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---



---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, then click OK.

## 7.7.6 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data

Perform the procedures in this section if you are upgrading from Oracle BI Applications release 7.9.6 to release 7.9.6.1.

---



---

**Note:** This procedure is applicable only if you are using an Oracle database for your data warehouse database.

---



---

This section includes the following procedures:

- [Section 7.7.6.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.1"](#)
- [Section 7.7.6.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 7.7.6.3, "Verifying the Data Migrated Successfully"](#)

### 7.7.6.1 Upgrading the Data Warehouse Schema to Version 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

**To upgrade the data warehouse schema to version 7.9.6.1**

1. Run the UPGRADE\_7961.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

2. Run the 796ORA\_TO\_7961ORA\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
3. Run the ddl\_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 7.7.6.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Run the 7961\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.
2. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
4. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT88 and execute the first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
5. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
6. Run the 7961\_UPGRADE\_POST\_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

### 7.7.6.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
  - 7961\_UPGRADE\_DROP\_INDEXES.sql
  - 7961\_UPGRADE\_PRE\_CTL\_SCRIPT.sql
  - 7961\_UPGRADE\_PRE\_DIMENSION.sql
  - 7961\_UPGRADE\_POST\_SCRIPT.sql
4. Check the log files for the ddl\_7961.ctf and Upgrade\_7961.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as `W_xxxx_x_79x`, `LKP_xxxx_x` and `79x_xxxx_TMP`.  
This step frees the space occupied by these backup tables.
6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 7.7.7 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.1 to 7.9.6.2.

This section contains the following procedures:

- [Section 7.7.7.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 7.7.7.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 7.7.7.3, "Verifying the Data Migrated Successfully"](#)

### 7.7.7.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

#### To upgrade the data warehouse schema to version 7.9.6.2

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the `7962_UPGRADE_PRE_UPG_CTL_SCRIPT.sql` file. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\SQLServer`.
4. Run the `UPGRADE_7962.ctl` script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)



5. Run the 7962\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the DDL\_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 7.7.7.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Run the 7962\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
- 3. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
- 4. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)
- 5. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
- 6. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
- 7. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
- 8. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT88 and execute the first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
- 10. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

### 7.7.7.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.

2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql

7962\_UPGRADE\_PRE\_CTL\_SCRIPT.sql

7962\_UPGRADE\_PRE\_DIMENSION.sql

createETLIndexes.sql

createQueryIndexes.sql

4. Check the log files for the DDL\_7962.ctf and Upgrade\_7962.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---



---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---



---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 7.7.8 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Upgrading the Data Warehouse Schema to Version 7.9.6.3](#)
- [Migrating Data into the Upgraded Data Warehouse](#)
- [Verifying the Data Migrated Successfully](#)

### 7.7.8.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

#### To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)  
**Note:** You must drop the query indexes before running the scripts to upgrade the schema.
3. If you are using a SQL Server database, run the 7963\_UPGRADE\_PRE\_UPG\_CTL\_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE\_7963.ctf script. For instructions, see [Section E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the DDL\_7963.ctf script. For instructions, see [Section E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 7.7.8.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Run the 7963\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Run the 7963\_UPGRADE\_DROP\_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 3. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Section D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
- 4. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
- 5. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
- 6. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
- 7. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)

8. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

9. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\

If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
10. Run the 7963\_UPGRADE\_CREATE\_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 11. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
- 12. Run the createQueryIndices.sql script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

### 7.7.8.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
  - 7963\_UPGRADE\_PRE\_CTL\_SCRIPT.sql
  - 7963\_UPGRADE\_PRE\_DIMENSION.sql
  - 7963\_UPGRADE\_DROP\_ETLINDICES.sql
  - 7963\_UPGRADE\_CREATE\_ETLINDICES.sql
  - dropQueryIndices.sql
  - createQueryIndices.sql
4. Check the log files for the ddl\_7963.ctl and Upgrade\_7963.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_XXXX\_X\_79X, LKP\_XXXX\_X and 79X\_XXXX\_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 7.8 Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases

If your data warehouse database platform is Teradata, you can upgrade to Oracle BI Applications version 7.9.6.3 from Oracle BI Applications 7.9.x.

This section includes the following topics:

- [Section 7.8.1, "Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3"](#)
- [Section 7.8.2, "Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3"](#)
- [Section 7.8.3, "Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3"](#)
- [Section 7.8.4, "Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3"](#)
- [Section 7.8.5, "Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3"](#)
- [Section 7.8.6, "Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3"](#)

### 7.8.1 Upgrading Oracle BI Applications From Version 7.9.3 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.3 to version 7.9.6.3 if your data warehouse database platform is Teradata.

**To upgrade the data warehouse schema and migrate data for a Teradata database**

1. Execute the following scripts:
  - 794\_UPGRADE\_TMP\_TABLES.sql
  - 795\_UPGRADE\_TMP\_TABLES.sql

- 7951\_UPGRADE\_TMP\_TABLES.sql
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 794\_RENAME\_AND\_EXCEPTIONS.sql script.
- a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 794\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
- ```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
- d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
 - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
 - f. Execute the 794_RENAME_AND_EXCEPTIONS.sql script.
 - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
- a. Log into DAC.
 - b. From the menu bar, select Tools, then ETL Management, and then Configure.
 - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
- The Configuration tab of the Data Warehouse Configuration Wizard is active.
- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
- The Data Warehouse SQL tab is active.
- e. Do the following:
 - Select the appropriate physical data source name for the Teradata database.
 - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

upgrade-regular.sql. This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

upgrade-questionable.sql. This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

upgradedwtables_sql.log. This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

upgrade-issues.log. This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.


```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
 - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
 - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
 - i. Execute the scripts in the upgrade-regular.sql file.
 - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
 - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
 - 794_UPDATE_DEFAULTS.sql
 - 795_UPDATE_DEFAULTS.sql
 - 7951_UPDATE_DEFAULTS.sql
 - 7961_UPDATE_DEFAULTS.sql
 - 7962_UPDATE_DEFAULTS.sql
 - 7963_UPDATE_DEFAULTS.sql
 6. Execute the 794_Backup Tables.sql script.
 7. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)

8. Configure the Informatica parameter file 794_UPG_PARAMS.txt. For instructions, see [Appendix A.1.2, "Setting Parameters in 794_UPG_PARAMS.txt."](#)
9. In Informatica Workflow Manager, navigate to the folder UPGRADE_793_to_794_PSFT88 and execute the UPGRADE_DIMENSIONS and UPGRADE_FACTS workflows.
10. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
11. Execute the 795_Backup Tables.sql script.
12. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
13. Configure the Informatica parameter file 795_UPG_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795_UPG_PARAMS.txt."](#)
14. In Informatica Workflow Manager, navigate to the folder UPGRADE_794_to_795_PSFT88 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
15. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
16. Execute the 7951_Backup Tables.sql script.
17. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)

- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
18. Configure the Informatica parameter file 7951_UPG_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951_UPG_PARAMS.txt."](#)
 19. In Informatica Workflow Manager, navigate to the folder UPGRADE_795_to_7951_PSFT and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
 20. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
 21. Execute the 796_Backup Tables.sql script.
 22. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
 23. Configure the Informatica parameter file 796_UPG_PARAMS.txt. For instructions, see [Appendix A.1.5, "Setting Parameters in 796_UPG_PARAMS.txt."](#)
 24. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
 25. If you are upgrading Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
 26. Back up and truncate the table W_POSITION_DH.
 27. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_PSFT88 and execute the following workflows in the order as they appear: UPGRADE_DIMENSIONS, UPGRADE_FACTS, and RESET_DAC_REFRESH_DATES.

If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE_7951_to_796_PSFT90 and execute the following workflows in the order as they appear: UPGRADE_DIMENSIONS, UPGRADE_FACTS, and RESET_DAC_REFRESH_DATES.

28. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the `7961_Backup Tables.sql` script.
30. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file `7961_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961_UPG_PARAMS.txt."](#)
32. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_PSFT88` and execute and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.

 If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_PSFT90` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
33. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
34. Execute the `7962_Backup Tables.sql` script.
35. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset_infa_seq_gen.bat on Upgrade Informatica Repository."](#)
36. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962_UPG_PARAMS.txt."](#)
37. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962_UPG_PARAMS.txt for PeopleSoft Source Systems."](#)

38. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962_UPG_PARAMS.txt for Oracle Financial Analytics."](#)
39. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
40. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_PSFT88 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.

If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE_7961_to_7962_PSFT90 and execute first the UPGRADE_DIMENSIONS workflow and then the UPGRADE_FACTS workflow.
41. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle_BI_DW_Base.rep).
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
42. Execute the 7963_Backup_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
43. Use the reset_infa_seq_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
 - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset_infa_seq_gen_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
 - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset_infa_seq_gen_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
44. Configure the Informatica parameter file 7963_UPG_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963_UPG_PARAMS.txt."](#)
45. If you are upgrading Oracle Financial Analytics, do the following:
 - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
 - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
46. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Human Resources Analytics."](#)
47. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963_UPG_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)

48. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_PSFT88` and execute first the `UPGRADE_DIMENSIONS` workflow, and then the `UPGRADE_FACTS` workflow.

If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7962_to_7963_PSFT90` and execute first the `UPGRADE_DIMENSIONS` workflow, and then the `UPGRADE_FACTS` workflow.

49. Upgrade geography-related attributes if necessary for your environment:
- Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- If the query returns a count greater than zero, run the `7963_UPGRADE_POST_SCRIPT.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\<database type>`.
If the query returns a count of zero, you do not need to run the `7963_UPGRADE_POST_SCRIPT.sql` script.
50. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset_infa_seq_gen_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
- If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset_infa_seq_gen_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

51. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- In the Informatica Administrator, select the Upgrade repository service.
 - In the General Properties area of the Properties tab, click Edit.
 - Make sure the operating mode of the repository service is set to Exclusive.
 - Click OK.
 - Choose Actions, and then click Delete Contents.
 - In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

7.8.2 Upgrading Oracle BI Applications From Version 7.9.4 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.4 to version 7.9.6.3 if your data warehouse database platform is Teradata.

To upgrade the data warehouse schema and migrate data for a Teradata database

- Execute the following scripts:
 - `795_UPGRADE_TMP_TABLES.sql`
 - `7951_UPGRADE_TMP_TABLES.sql`

- 7961_UPGRADE_TMP_TABLES.sql
 - 7962_UPGRADE_TMP_TABLES.sql
 - 7963_UPGRADE_TMP_TABLES.sql
 - a. Open the SQL client for your database type.
 - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - c. Open the <version>_UPGRADE_TMP_TABLES.sql file, and copy the contents into the SQL client.
 - d. Execute the script.
2. Edit and execute the 795_RENAME_AND_EXCEPTIONS.sql script.
- a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
 - b. Open the 795_RENAME_AND_EXCEPTIONS.sql script.
 - c. Search for the following statements to locate columns with RENAME commands.
- ```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
- d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 795\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
- a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.
- The Configuration tab of the Data Warehouse Configuration Wizard is active.
- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.
- The Data Warehouse SQL tab is active.
- e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
    - 795\_UPDATE\_DEFAULTS.sql
    - 7951\_UPDATE\_DEFAULTS.sql
    - 7961\_UPDATE\_DEFAULTS.sql
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  6. Execute the 795\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  8. Configure the Informatica parameter file 795\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt."](#)

9. In Informatica Workflow Manager, navigate to the folder UPGRADE\_794\_to\_795\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
11. Execute the 7951\_Backup Tables.sql script.
12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
13. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
14. In Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_PSFT and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
15. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
16. Execute the 796\_Backup Tables.sql script.
17. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
18. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)

19. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
20. If you are upgrading Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
21. Back up and truncate the table W\_POSITION\_DH.
22. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT88 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT90 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.
23. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
24. Execute the 7961\_Backup Tables.sql script.
25. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
27. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
28. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).



- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
29. Execute the 7962\_Backup\_Tables.sql script.
30. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
31. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
32. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)
33. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
34. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
35. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
36. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
37. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
38. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
39. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
40. If you are upgrading Oracle Financial Analytics, do the following:
- a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
41. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
42. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)
43. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
- If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
44. Upgrade geography-related attributes if necessary for your environment:
- a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
45. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

46. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

### 7.8.3 Upgrading Oracle BI Applications From Version 7.9.5 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.5 to version 7.9.6.3 if your data warehouse database platform is Teradata.

#### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 7951\_UPGRADE\_TMP\_TABLES.sql
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 7951\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the

teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.

4. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.

5. Execute the following scripts:
  - 7951\_UPDATE\_DEFAULTS.sql
  - 7961\_UPDATE\_DEFAULTS.sql
  - 7962\_UPDATE\_DEFAULTS.sql
  - 7963\_UPDATE\_DEFAULTS.sql
6. Execute the 7951\_Backup Tables.sql script.
7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
8. Configure the Informatica parameter file 7951\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt."](#)
9. In Informatica Workflow Manager, navigate to the folder UPGRADE\_795\_to\_7951\_PSFT and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
10. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
11. Execute the 796\_Backup Tables.sql script.
12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
13. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
14. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)

15. If you are upgrading Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
16. Back up and truncate the table W\_POSITION\_DH.
17. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT88 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT90 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.
18. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
19. Execute the 7961\_Backup\_Tables.sql script.
20. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
21. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
22. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
23. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

24. Execute 7962\_Backup\_Tables.sql script.
25. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
27. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)
28. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
29. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
30. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
31. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
32. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
33. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)

34. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
35. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
36. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
37. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)
38. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
39. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
40. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
41. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.



- d. Click OK.
- e. Choose Actions, and then click Delete Contents.
- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 7.8.4 Upgrading Oracle BI Applications From Version 7.9.5.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.5.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 7961\_UPGRADE\_TMP\_TABLES.sql
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. Edit and execute the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - a. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - b. Open the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - c. Search for the following statements to locate columns with RENAME commands.
 

```
ALTER TABLE <Table_Name> RENAME <Column Name> TO <New Column Name>;
```
  - d. For all columns with RENAME commands, check to see whether any COLUMN STATISTICS are defined on them.
  - e. For all columns that have STATISTICS defined on them, back up the COLUMN STATISTICS definition and then drop them.
  - f. Execute the 7961\_RENAME\_AND\_EXCEPTIONS.sql script.
  - g. Recreate STATISTICS on any column on which they were dropped.
3. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
4. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.

- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
5. Execute the following scripts:
    - 7961\_UPDATE\_DEFAULTS.sql
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  6. Execute the 796\_Backup Tables.sql script.
  7. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
8. Configure the Informatica parameter file 796\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt."](#)
  9. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
  10. If you are upgrading Oracle Project Analytics, you need to configure specific parameters. For more information, see [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics."](#)
  11. Back up and truncate the table W\_POSITION\_DH.
  12. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT88 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7951\_to\_796\_PSFT90 and execute the following workflows in the order as they appear: UPGRADE\_DIMENSIONS, UPGRADE\_FACTS, and RESET\_DAC\_REFRESH\_DATES.
  13. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
  14. Execute the 7961\_Backup Tables.sql script.
  15. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  16. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)

17. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_PSFT88` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder `UPGRADE_796_to_7961_PSFT90` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
18. Execute the `7962_Backup_Tables.sql` script.
19. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
20. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
21. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)
22. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
23. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
24. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_PSFT88` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_PSFT90` and execute first the `UPGRADE_DIMENSIONS` workflow and then the `UPGRADE_FACTS` workflow.
25. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
26. Execute the `7963_Backup_Tables.sql` script. This file is located in `OracleBI\dwrep\Upgrade\DbScripts\Teradata`.
27. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
- 28. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
- 29. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
- 30. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
- 31. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)
- 32. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
- 33. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
- 34. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)

35. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 7.8.5 Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.1 to version 7.9.6.3 if your data warehouse database platform is Teradata.

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following scripts:
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.  
The Configuration tab of the Data Warehouse Configuration Wizard is active.
  - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.  
The Data Warehouse SQL tab is active.
  - e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the following scripts:
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  5. Execute the 7962\_Backup\_Tables.sql script.
  6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  7. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
  8. Configure additional parameters specific to PeopleSoft source systems. For instructions, see [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)

9. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For instructions, see [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
10. If you are upgrading Oracle Financial Analytics, you need to make sure specific indexes are created. For instructions, see [Appendix C, "Creating Indexes for Oracle Financial Analytics."](#)
11. If you are using PeopleSoft version 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft version 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
12. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
13. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
14. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
15. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
16. If you are upgrading Oracle Financial Analytics, do the following:
  - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
17. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
18. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)



19. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.

If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.

20. Upgrade geography-related attributes if necessary for your environment:
- a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
21. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
- If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
22. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
- a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, then click OK.

## 7.8.6 Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3

Follow this procedure to upgrade from Oracle Business Applications version 7.9.6.2 to version 7.9.6.3 if your data warehouse database platform is Teradata

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the 7963\_UPGRADE\_TMP\_TABLES.sql script.
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.

- c. Open the 7963\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.

3. Generate upgrade scripts from DAC.

- a. Log into DAC.
- b. From the menu bar, select Tools, then ETL Management, and then Configure.
- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.

- h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the 7963\_UPDATE\_DEFAULTS.sql scripts.
  5. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  7. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
  8. If you are upgrading Oracle Financial Analytics, do the following:
    - a. Set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
    - b. Set the parameters as specified in [Appendix A.3.5.2, "Setting Parameters Specific to PeopleSoft."](#)
  9. If you are upgrading Oracle Human Resources, set parameters as specified in [Appendix A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics."](#)
  10. If your source system is PeopleSoft version 9.0 and you are upgrading Oracle Procurement and Spend Analytics, set the parameter as specified in [Appendix A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System."](#)
  11. If you are using PeopleSoft 8.8, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT88 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.  
  
If you are using PeopleSoft 8.9 or 9.0, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_PSFT90 and execute first the UPGRADE\_DIMENSIONS workflow, and then the UPGRADE\_FACTS workflow.
  12. Upgrade geography-related attributes if necessary for your environment:
    - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```



## 7.10 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server embedded LDAP server.

### To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, `myrealm`).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).
4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

## 7.11 Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and web catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Oracle BI Repository. This repository is the one that you originally received from Oracle as part of your current Oracle BI Applications installation (the version from which you are upgrading). Standard repositories from previous releases are available in the installation folder `$ORACLE_BI_HOME\biapps\upgrade`.
- The Oracle BI Repository you are currently using (the repository deployed in production).
- The Presentation Catalog you are currently using (the catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

---



---

**Note:** Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 7.9, "Localizing the Policy Store."](#)

---



---

## 7.12 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

---



---

**Note:** Before you begin this procedure, you must have already upgraded the Oracle BI repository you are currently using and the standard (out-of-the-box) Oracle BI repository version 7.9.x (the version from which you are upgrading) to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---



---

This process merges your customizations of a prior release of the Oracle BI repository with the new release of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository.

To upgrade the repository, perform the following tasks:

- [Section 7.12.1, "Preparing for the Oracle BI Repository Upgrade"](#)
- [Section 7.12.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 7.12.3, "Comparing the Oracle BI Repositories"](#)
- [Section 7.12.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 7–1](#) provides the names and descriptions of the repositories used in the examples in this section.

**Table 7–1 Names of Analytics Repositories used in Examples**

| Name of Repository            | Description                                                                                                                                                                                             |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OracleBIAnalyticsApps_79x.rpd | The standard Oracle BI repository for the version you are upgrading from.<br><br><b>Note:</b> Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade. |

**Table 7–1 (Cont.) Names of Analytics Repositories used in Examples**

| Name of Repository                     | Description                                                                                        |
|----------------------------------------|----------------------------------------------------------------------------------------------------|
| OracleBIAnalyticsApps.rpd              | The standard Oracle BI repository for the version you are upgrading to.                            |
| Customer_<br>OracleBIAnalyticsApps.rpd | The Oracle BI repository that contains your customizations for the version you are upgrading from. |
| Merged_Repository_OracleBI.rpd         | The Oracle BI repository that contains your customizations for the version you are upgrading to.   |

## 7.12.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

### To prepare for the Analytics repository upgrade

- Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
  - Original
  - AfterEqualize
  - AfterMerge
  - AfterManualWork
  - AfterRegressions
- Copy the following repositories to the folder \OracleBIUpgrade\Original:
  - The standard repository that you upgraded to the 11g format in [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#) (for example, OracleBIAnalyticsApps\_79x.rpd).
  - The production repository that you also upgraded to the 11g format in [Section 7.11](#) (for example, Customer\_OracleBIAnalyticsApps.rpd).
  - The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Oracle BI Applications for one or more modules using a Oracle BI repository in which you extracted the corresponding projects for the modules from the standard Oracle BI repository file you received from the previous release, you need to extract the same projects from the OracleBIAnalyticsApps\_79x.rpd file and use this as your original repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Oracle BI repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the OracleBIAnalyticsApps\_79x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

## 7.12.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories

(for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer\_OracleBIAnalyticsApps.rpd whether it is coming from the OracleBIAnalyticsApps\_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W\_DAY\_D" might be considered equivalent to "Core"."W DAY D" if the characters "\_" and " " have been declared as equivalent.
- **Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the equalizerpds executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

[Table 7–2](#) provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

**Table 7–2 Rename MAP Files to Be Used for Various Releases**

| <b>Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)</b> | <b>Rename MAP File to Be Used</b> |
|-----------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Siebel Business Analytics Applications 7.0.x                                                                    | Not available                     |
| Siebel Business Analytics Applications 7.5.x                                                                    | Not available                     |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)                                   | Rename77-7963.map                 |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)                                       | Rename771-7963.map                |
| Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release                         | Rename782-7963.map                |
| Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release                          | Rename783-7963.map                |
| Oracle BI Applications 7.9.0                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.1                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.2                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.3                                                                                    | Rename793to7963.map               |
| Oracle BI Applications 7.9.4                                                                                    | Rename794to7963.map               |
| Oracle BI Applications 7.9.5                                                                                    | Rename79x-7963.map                |



**Table 7–2 (Cont.) Rename MAP Files to Be Used for Various Releases**

| <b>Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)</b> | <b>Rename MAP File to Be Used</b> |
|-----------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Oracle BI Applications 7.9.5.1                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.5.2                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.6                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.6.2                                                                                  | Not required                      |

The syntax of the equalizerpds command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udml_utf8_file_name_equalization] [-O output_repository_name]
[-Y equalStringSet]
-Y Treat the characters as equals.
/? Display this usage information and exit.
```

### To equalize a repository

1. Copy the appropriate MAP file from the \$ORACLE\_BI\_HOME\biapps\upgrade folder into the folder where you will execute equalizerpds.exe, for example, \OracleBIUpgrade\Original.
2. Run equalizerpds.exe to equalize the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsAppss\_793.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder.

If the equalizerpds.exe executable file runs correctly, no errors are returned.

3. Run equalizerpds.exe to equalize your customized repository (for example, Customer\_OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsApps\_79x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_OracleBIAnalyticsApps.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_OracleBIAnalyticsApps.rpd
```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.

Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

### 7.12.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

#### To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

### 7.12.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

#### To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, OracleBIAnalyticsApps\_79x.rpd).
6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.
9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, Merged\_Repository\_OracleBIAnalyticsApps.rpd.  
The new repository will contain the final results of the merge.
10. Select the **Equalize during merge** check box.
11. Click **Next**.
12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.  
When all rows have a value in the Decision field, the Finish button is enabled.
14. Select the **Check consistency of the merged RPD** check box.
15. Click **Finish**.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

17. Copy the repository to the folder \OracleBIUpgrade\AfterManualWork.

## 7.13 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

You need to perform the following tasks to complete the presentation merge process.

- [Section 7.13.1, "Trimming the Input Presentation Catalog"](#)
- [Section 7.13.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 7.13.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This catalog is the output from the step you performed in [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the version of the unmodified presentation catalog you received with the Oracle BI Applications release you are upgrading from that you upgraded to the 11g format.
- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This catalog is the output from the step you performed in [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

[Catalog to the Oracle BI Enterprise Edition 11g Version.](#)" It is the presentation catalog currently in use at your organization that you upgraded to the 11g format.

---

---

**Caution:** In releases of Oracle BI Applications previous to 7.9, the Presentation Catalog (formerly known as the Siebel Analytics Web Catalog) was stored in a single file rather than in a directory structure of individual files. If you have a previous version of the Presentation Catalog, you will need to convert it to the new format. For more information about how to convert the Presentation Catalog to the new format, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*

---

---

### 7.13.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

### 7.13.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 7.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

---

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

#### To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.

For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.

3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name>
-oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group
-new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

**Note:** If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

### 7.13.3 Testing the Results of the Presentation Catalog Upgrade

---



---

**Note:** Before you perform this step, you must first migrate the data into the upgraded data warehouse.

---



---

The Presentation Catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the Presentation Catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged Presentation Catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new Presentation Catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

## 7.14 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

### To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the Presentation Catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE\_BI\_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.
5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAAnalyticsApps.rpd and the production repository.

# Part IV

---

## Upgrading When Your Source System is JD Edwards

Part IV contains instructions for upgrading to the current release of Oracle BI Applications when you are running JD Edwards Enterprise as your source system.

Part IV contains [Chapter 8, "Upgrading Oracle BI Applications for JD Edwards Source Systems."](#)





---

---

# Upgrading Oracle BI Applications for JD Edwards Source Systems

---

---

**Note:** Some of the information about database platforms and source systems might not apply to this version of Oracle Business Intelligence Applications. For up-to-date information about supported databases and source systems in this version of Oracle Business Intelligence Applications, make sure you read *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*. Make sure that you also read the *Oracle Business Intelligence Applications Release Notes*. The most up-to-date versions of these documents are located on the Oracle Technology Network at <http://www.oracle.com/technetwork/middleware/bi-foundation/documentation/index.html>. To register for a free account on the Oracle Technology Network, go to <http://www.oracle.com/technetwork/index.html>.

---

---

This section contains instructions for upgrading Oracle BI Applications when you are running JD Edwards Enterprise Applications as your source system

This section contains the following topics:

- [Section 8.1, "Upgrading Oracle BI Infrastructure"](#)
- [Section 8.2, "Upgrading Oracle BI Applications"](#)
- [Section 8.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2"](#)
- [Section 8.4, "Upgrading the Informatica Repository"](#)
- [Section 8.5, "Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC"](#)
- [Section 8.6, "Upgrading and Configuring DAC"](#)
- [Section 8.7, "Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases"](#)
- [Section 8.8, "Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases"](#)
- [Section 8.9, "Localizing the Policy Store"](#)
- [Section 8.10, "Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server"](#)

- [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#)
- [Section 8.12, "Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository"](#)
- [Section 8.13, "Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 8.14, "Regression Testing the Oracle BI Repository Merge"](#)

## 8.1 Upgrading Oracle BI Infrastructure

Upgrade the Oracle BI Infrastructure to the version that is supported for this release of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for the current version that is supported. For information on installing the supported version of Oracle BI Infrastructure, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

## 8.2 Upgrading Oracle BI Applications

Run the Oracle BI Applications installer to upgrade your Oracle BI Applications environment to the current version. For instructions on running the installer, see *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

Note the following points:

- Review "Chapter 3: Preinstallation and Predeployment Requirements for Oracle BI Applications," in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*. You must satisfy these requirements in order to upgrade to the current version of Oracle BI Applications.
- Perform only the step that addresses the running of the installer. Do not perform the configuration instructions that follow the running of the installer.
- If you have a previous release of Oracle BI Applications installed, you must uninstall it before you run the installer for the current release. If you do not uninstall the old release, some folders from the current release will not be correctly installed. (Make a back-up of your DAC folder before you uninstall the old release. This will be the backup of your DAC Client and Server and the DAC metadata files.)

## 8.3 Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2

You must install Informatica PowerCenter 9.0.1 Hotfix 2 to run the current version of Oracle BI Applications. See the *System Requirements and Supported Platforms for Oracle Business Intelligence Applications* for information about additional hotfixes or emergency bug fixes that may be required to support the current version of Informatica.

The components and architecture for Informatica PowerCenter 9.0.1 differ significantly from previous versions. Oracle recommends that you carefully review the Informatica PowerCenter 9.0.1 documentation, which is included on the Informatica DVD provided with Oracle BI Applications.

For a summary of installation instructions for installing Informatica PowerCenter 9.0.1 on a single machine in an Oracle BI Applications deployment, see the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*.

For detailed information about deploying Informatica PowerCenter 9.0.1, refer to the *Informatica PowerCenter Installation and Configuration Guide*, *Informatica Upgrade Guide for PowerCenter 8.1.x*, *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, *Informatica PowerCenter Administrator Guide*, and related documentation.

### To upgrade to Informatica PowerCenter 9.0.1 Hotfix 2

- Perform the pre-upgrade, upgrade, and post-upgrade steps documented in *Informatica Upgrade Guide for PowerCenter 8.1.x* or *Informatica Upgrade Guide for PowerCenter 8.5.x and 8.6.x*, depending on which version of Informatica PowerCenter you are upgrading from.

Note the following points:

- The Informatica PowerCenter 9.0.1 installation process includes upgrading your current Informatica Repository to the version 9.0.1 format. This process is necessary so that you will be able to access your current repository using version 9.0.1 client tools and so that you can perform the procedure [Section 8.4, "Upgrading the Informatica Repository."](#)

In [Section 8.4, "Upgrading the Informatica Repository,"](#) you back up and rename your current repository and then restore the Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep) that is installed during the Oracle BI Applications installation. You then copy your custom folder from the backed up repository into the newly restored Oracle\_BI\_DW\_Base repository.

- If you install Informatica PowerCenter 9.0.1 using the installer Upgrade option rather than the Install option, the property \$PMRootDir is not updated to the upgraded service version installation directory. You need to configure this property to specify the upgraded service installation directory in order for DAC to be able to locate the Informatica source files.

To configure the \$PMRootDir property: launch the Informatica Administrator, select the Integration Service from the Navigator, click the Processes tab, click Edit, and enter the appropriate value for \$PMRootDir.

## 8.4 Upgrading the Informatica Repository

Follow this procedure to upgrade the Informatica Repository.

For detailed instructions on backing up and restoring the Informatica Repository, see the topic titled, "Backing Up and Restoring the Repository," in "Chapter 15: PowerCenter Repository Management," in the *Informatica PowerCenter Administrator Guide*, which is included on the Informatica DVD provided with Oracle BI Applications.

### To upgrade the Informatica Repository

1. Make sure you have backed up and renamed your current Informatica Repository.

---



---

**Note:** This repository must be upgraded to the version 9.0.1 format during the procedure in [Section 8.3, "Upgrading to Informatica PowerCenter Version 9.0.1 Hotfix 2."](#) You must upgrade this repository to the version 9.0.1 format in order to move your custom folder from this repository into the new Oracle\_BI\_DW\_Base repository that you restore in the steps below.

---



---

2. Copy the Oracle\_BI\_DW\_Base.rep file from the folder OracleBI\dwrep\Informatica\Repository into the folder <INFA\_HOME>\server\infa\_shared\Backup.

---

**Note:** The Oracle\_BI\_DW\_Base.rep file is installed in the OracleBI root directory when you run the Oracle BI Applications installer, as described in [Section 8.2, "Upgrading Oracle BI Applications."](#)

---

3. Restore the Oracle\_BI\_DW\_Base.rep repository into an empty schema.
4. Copy the custom folder from your previous Informatica Repository to the newly created Informatica Repository:
  - a. Launch the Informatica PowerCenter Repository Manager, and connect to both your previous and newly created Informatica repositories.
  - b. Copy the Custom folder in your previous repository into the newly created Informatica Repository.
  - c. Make sure there is an individual workflow for each of the mappings in the Custom folder.

## 8.5 Configuring Informatica PowerCenter Version 9.0.1 to Work with Oracle BI Applications and DAC

Informatica PowerCenter version 9.0.1 requires additional configuration steps to work with Oracle BI Applications and DAC. For instructions on performing these steps, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Copying Source Files and Lookup Files
- Setting PowerCenter Integration Services Relaxed Code Page Validation
- Setting PowerCenter Integration Services Custom Properties
- Creating the Repository Administrator User in the Native Security Domain

## 8.6 Upgrading and Configuring DAC

This section includes information you must follow to upgrade and configure DAC. It includes the following topics:

- [Section 8.6.1, "Installing the DAC Platform and Oracle BI Applications Metadata Repository Files"](#)
- [Section 8.6.2, "Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica"](#)
- [Section 8.6.3, "Upgrading the DAC Repository"](#)

### 8.6.1 Installing the DAC Platform and Oracle BI Applications Metadata Repository Files

The current version of DAC is installed by its own installer and not the Oracle BI Applications installer. For information about the version of the DAC platform supported with the current release of Oracle BI Applications, see *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*.

After you install DAC, you then need to copy metadata files from the machine hosting Oracle BI Applications to the machines hosting the DAC Client and Server. You then need to import the new metadata into the DAC Repository. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Installing and Setting Up the DAC Platform

---



---

**Note:** You must perform all of the procedures in this section, including installing JDBC drivers, creating ODBC database connections, and installing pmrep and pmcmd command line programs.

---



---

- Installing DAC Metadata Files
- Logging into DAC for the First Time and Importing Metadata into the DAC Repository

After you complete this procedure, you will have the default DAC Repository for Oracle BI Applications version 7.9.6.3.

## 8.6.2 Configuring the DAC Client and Server to Work with Oracle BI Applications and Informatica

You need to perform certain configuration tasks to enable the DAC Client and Server to work with Oracle BI Applications and Informatica PowerCenter. For instructions on performing these tasks, see the following sections in the *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*:

- Configuring the DAC Server
- Configuring DAC Integration Settings
- Configuring the SiebelUnicodeDB Custom Property

---



---

**Note:** This procedure is required only if your source to target data movement configuration is Unicode to Unicode.

---



---

- Setting Up DAC to Receive Email Notification
- Additional Configuration Tasks

In this section, perform only the tasks that apply to your environment.

## 8.6.3 Upgrading the DAC Repository

You will use the Refresh Base option of the DAC's Upgrade/Merge Wizard to complete the upgrade of your existing DAC Repository. For information about how to use the Refresh Base option of the Upgrade/Merge Wizard, see the topic titled, "About the Refresh Base Option," in "Chapter 10: Upgrading, Comparing and Merging DAC Repositories," in the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

The *Oracle Business Intelligence Data Warehouse Administration Console User's Guide* is available in the Oracle Business Intelligence Data Warehouse Administration Console Documentation Library on the Oracle Technology Network.

Before you upgrade the DAC Repository, do the following:

- Make a copy of your default source system container. (You cannot change the metadata in the preconfigured container. You must make a copy of it in order to be able to modify the objects it contains.) For instructions, see the *Oracle Business Intelligence Data Warehouse Administration Console User's Guide*.

## 8.7 Upgrading the Data Warehouse Schema and Migrating Data for Non-Teradata Databases

Perform the procedures in this section if you are upgrading from Oracle BI Applications release 7.9.6, 7.9.6.1, or 7.9.6.2 to release 7.9.6.3.

This section includes the following procedures:

- [Section 8.7.1, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases"](#) (Applicable only if you are using an Oracle database as your data warehouse database.)
- [Section 8.7.2, "Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases"](#)
- [Section 8.7.3, "Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases"](#)

If you are currently running Oracle BI Applications release 7.9.6, you need to perform the procedures in [Section 8.7.1](#), [Section 8.7.2](#), and [Section 8.7.3](#).

If you are currently running Oracle BI Applications release 7.9.6.1, you need to perform the procedures in [Section 8.7.2](#) and [Section 8.7.3](#).

If you are currently running Oracle BI Applications release 7.9.6.2, you need to perform the procedures in [Section 8.7.3](#).

### 8.7.1 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1 and Migrating Data for Non-Teradata Databases

Follow the procedures in this section to upgrade the data warehouse schema to version 7.9.6.1 and migrate data.

---



---

**Note:** This procedure is applicable only if you are using an Oracle database for your data warehouse database.

---



---

This section includes the following topics:

- [Section 8.7.1.1, "Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1"](#)
- [Section 8.7.1.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 8.7.1.3, "Verifying the Data Migrated Successfully"](#)

#### 8.7.1.1 Upgrading the Data Warehouse Schema From Version 7.9.6 to 7.9.6.1

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.1.

**To upgrade the data warehouse schema to version 7.9.6.1**

1. Run the UPGRADE\_7961.ctl script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

2. Run the 796ORA\_TO\_7961ORA\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\Oracle.
3. Run the ddl\_7961.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 8.7.1.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Run the 7961\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.
2. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
3. Configure the Informatica parameter file 7961\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt."](#)
4. If you are upgrading Oracle Financial Analytics, you need to configure specific parameters and update mappings. For more information, see [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)
5. If you are using JD Edwards version 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_JDE\_811SP1\_812 and execute the first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

If you are using JD Edwards version 9, in Informatica Workflow Manager, navigate to the folder UPGRADE\_796\_to\_7961\_JDE\_90 and execute first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.

6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
7. Run the 7961\_UPGRADE\_POST\_SCRIPT.sql. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Oracle.

### 8.7.1.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

7961\_UPGRADE\_DROP\_INDEXES.sql

796ORA\_TO\_7961ORA\_UPGRADE\_PRE\_CTL\_SCRIPT.sql

7961\_UPGRADE\_PRE\_DIMENSION.sql

7961\_UPGRADE\_POST\_SCRIPT.sql

4. Check the log files for the ddl\_7961.ctf and Upgrade\_7961.ctf scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.

This step frees the space occupied by these backup tables.

## 8.7.2 Upgrading the Data Warehouse Schema From Version 7.9.6.1 to 7.9.6.2 and Migrating Data for Non-Teradata Databases

This section contains the following procedures:

- [Section 8.7.2.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.2"](#)
- [Section 8.7.2.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 8.7.2.3, "Verifying the Data Migrated Successfully"](#)
- [Section 8.7.2.4, "Resetting Refresh Dates"](#)

### 8.7.2.1 Upgrading the Data Warehouse Schema to Version 7.9.6.2

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.2.

**To upgrade the data warehouse schema to version 7.9.6.2**

1. Generate the scripts to drop and create indexes. For instructions, see [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)
2. Run the script to drop all indexes. For instructions, see [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2."](#)
3. If you are using a SQL Server database, run the 7962\_UPGRADE\_PRE\_UPG\_CTL\_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE\_7962.ctf script. [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7962\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the DDL\_7962.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 8.7.2.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

**To migrate data into the upgraded data warehouse**



1. Run the 7962\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
- 3. Configure the Informatica parameter file 7962\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
- 4. Run the script to create ETL indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)
- 5. In Informatica Workflow Manager, navigate to the folder UPGRADE\_7961\_to\_7962\_JDE and execute the first the UPGRADE\_DIMENSIONS workflow and then the UPGRADE\_FACTS workflow.
- 6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
- 7. Run the script to create Query indexes. For instructions, see [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2."](#)

### 8.7.2.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:

dropallindexes.sql

7962\_UPGRADE\_PRE\_CTL\_SCRIPT.sql

7962\_UPGRADE\_PRE\_DIMENSION.sql

createETLIndexes.sql

createQueryIndexes.sql

4. Check the log files for the ddl\_7962.ctl and Upgrade\_7962.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as `W_xxxx_x_79x`, `LKP_xxxx_x` and `79x_xxxx_TMP`.

This step frees the space occupied by these backup tables.

#### 8.7.2.4 Resetting Refresh Dates

After verifying the data was migrated successfully into the upgraded data warehouse, follow this procedure to reset refresh dates.

1. If you are using JD Edwards version 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_JDE_811SP1_812` and execute the `RESET_DAC_REFRESH_DATES` workflow.

If you are using JD Edwards version 9, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_JDE90` and execute the `RESET_DAC_REFRESH_DATES` workflow.

2. Locate the `7962_RESET_REFRESH_DATES.txt` file in the folder `OracleBI\dwrep\Upgrade\Informatica\SrcFiles`.

In the next step, you will reset the refresh dates to NULL for the tables listed in `7962_RESET_REFRESH_DATES.txt`.

3. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the tables listed in the `7962_RESET_REFRESH_DATES.txt` file, and, for each table, set the value in the Refresh Date column to NULL
4. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

### 8.7.3 Upgrading the Data Warehouse Schema From Version 7.9.6.2 to 7.9.6.3 and Migrating Data for Non-Teradata Databases

Perform the procedures in this section to upgrade from Oracle BI Applications release 7.9.6.2 to 7.9.6.3.

This section contains the following procedures:

- [Section 8.7.3.1, "Upgrading the Data Warehouse Schema to Version 7.9.6.3"](#)
- [Section 8.7.3.2, "Migrating Data into the Upgraded Data Warehouse"](#)
- [Section 8.7.3.3, "Verifying the Data Migrated Successfully"](#)

### 8.7.3.1 Upgrading the Data Warehouse Schema to Version 7.9.6.3

Follow this procedure to upgrade the data warehouse schema to version 7.9.6.3.

#### To upgrade the data warehouse schema to version 7.9.6.3

1. Generate the scripts to drop and create query indexes. For instructions, see [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)
2. Run the script to drop query indexes. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)  
**Note:** You must drop the query indexes before running the scripts to upgrade the schema.
3. If you are using a SQL Server database, run the 7963\_UPGRADE\_PRE\_UPG\_CTL\_SCRIPT.sql file. This file is located in OracleBI\dwrep\Upgrade\DbScripts\SQLServer.
4. Run the UPGRADE\_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)
5. Run the 7963\_UPGRADE\_PRE\_CTL\_SCRIPT.sql. This file is located in the folder OracleBI\dwrep\Upgrade\DbScripts\- 6. Run the DDL\_7963.ctf script. For instructions, see [Appendix E.1, "Using the DDLimp Utility to Run CTL Files."](#)

### 8.7.3.2 Migrating Data into the Upgraded Data Warehouse

Follow this procedure to migrate data into the upgraded data warehouse.

#### To migrate data into the upgraded data warehouse

1. Run the 7963\_UPGRADE\_PRE\_DIMENSION\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 2. Run the 7963\_UPGRADE\_DROP\_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 3. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade repository. For instructions, see [Appendix D.4, "Running reset\\_infa\\_seq\\_gen.bat on Upgrade Informatica Repository."](#)
- 4. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Appendix A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
- 5. If you are upgrading Oracle Financial Analytics, set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
- 6. If you are using JD Edwards 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE\_811SP1\_812 and execute the UPGRADE\_FACTS workflow.

If you are using JD Edwards 9, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE90 and execute the UPGRADE\_FACTS workflow.

7. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
```

```
COUNTRY LIKE '%' OR COUNTRY LIKE ' %' OR STATE_PROV LIKE '%' OR STATE_PROV
LIKE ' %' OR CITY LIKE '%' OR CITY LIKE ' %' OR ZIPCODE LIKE '%' OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
8. Run the 7963\_UPGRADE\_CREATE\_ETLINDICES.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\- 9. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep). For instructions, see [Appendix D.1, "Running reset\\_infa\\_seq\\_gen.bat on Base Informatica Repository."](#)
- 10. Run the createQueryIndices script. For instructions, see [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

### 8.7.3.3 Verifying the Data Migrated Successfully

Follow this procedure to verify that the data was migrated successfully into the upgraded data warehouse.

#### To verify the data migrated successfully

1. Validate the data in the upgraded data warehouse.
2. Check whether any of the Informatica mapping log files stored in the <INFA\_HOME>\server\infa\_shared\Sesslogs directory indicates errors or failures.
3. Check whether the following scripts that you ran in the SQL client of the database failed or errored out while executing:
  - 7963\_UPGRADE\_PRE\_CTL\_SCRIPT.sql
  - 7963\_UPGRADE\_PRE\_DIMENSION.sql
  - 7963\_UPGRADE\_DROP\_ETLINDICES.sql
  - 7963\_UPGRADE\_CREATE\_ETLINDICES.sql
  - dropQueryIndices.sql
  - createQueryIndices.sql
4. Check the log files for the ddl\_7963.ctl and Upgrade\_7963.ctl scripts, which you ran using the DDLimp command, to determine whether any of the runs failed.

If you did not detect any errors or failures in the steps above, then the data migration was successful.

---

**Note:** Ignore any error messages that indicate tables or indexes could not be dropped from the database. These error messages are displayed when the specified objects do not exist in the database. This can occur because each source system container holds a subset of all data warehouse tables

---

5. If the data migration was successful, drop the tables that were created during the upgrade process, such as W\_xxxx\_x\_79x, LKP\_xxxx\_x and 79x\_xxxx\_TMP.

This step frees the space occupied by these backup tables.

6. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 8.8 Upgrading the Data Warehouse Schema and Migrating Data for Teradata Databases

If your data warehouse database platform is Teradata, follow the procedures in this section to upgrade from Oracle BI Applications version 7.9.6.1 and 7.9.6.2 to version 7.9.6.3.

This section includes the following topics:

- [Section 8.8.1, "Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.2"](#)
- [Section 8.8.2, "Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3"](#)

### 8.8.1 Upgrading Oracle BI Applications From Version 7.9.6.1 to 7.9.6.2

Follow this procedure to upgrade from Oracle BI Applications version 7.9.6.1 to version 7.9.6.2 if your data warehouse database platform is Teradata.

**To upgrade the data warehouse schema and migrate data for a Teradata database**

1. Execute the following scripts:
  - 7962\_UPGRADE\_TMP\_TABLES.sql
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.

- c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.

The Configuration tab of the Data Warehouse Configuration Wizard is active.

- d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.

The Data Warehouse SQL tab is active.

- e. Do the following:

- Select the appropriate physical data source name for the Teradata database.
- In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```

- g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the following scripts:
    - 7962\_UPDATE\_DEFAULTS.sql
    - 7963\_UPDATE\_DEFAULTS.sql
  5. Execute the 7962\_Backup\_Tables.sql script.

6. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
7. Configure the Informatica parameter file `7962_UPG_PARAMS.txt`. For instructions, see [Appendix A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#)
8. If you are using JD Edwards version 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_JDE_811SP1_812` and execute the following workflows in the order specified:
  - `UPGRADE_DIMENSIONS`
  - `UPGRADE_FACTS`
  - `RESET_DAC_REFRESH_DATES`
9. If you are using JD Edwards version 9, in Informatica Workflow Manager, navigate to the folder `UPGRADE_7961_to_7962_JDE90` and execute the following workflows in the order specified:
  - `UPGRADE_DIMENSIONS`
  - `UPGRADE_FACTS`
  - `RESET_DAC_REFRESH_DATES`
10. Use the `reset_infa_seq_gen.bat` script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, `Oracle_BI_DW_Base.rep`).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
11. Locate the `7962_RESET_REFRESH_DATES.txt` file in the folder `OracleBI\dwrep\Upgrade\Informatica\SrcFiles`.

In the next step, you will reset the refresh dates to NULL for the tables listed in `7962_RESET_REFRESH_DATES.txt`.
12. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the tables listed in the `7962_RESET_REFRESH_DATES.txt` file, and, for each table, set the value in the Refresh Date column to NULL

13. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
14. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
15. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
16. If you are upgrading Oracle Financial Analytics, set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
17. If you are using JD Edwards 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE\_811SP1\_812 and execute the UPGRADE\_FACTS workflow.  
  
If you are using JD Edwards 9, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE90 and execute the UPGRADE\_FACTS workflow.
18. Upgrade geography-related attributes if necessary for your environment:
  - a. Determine whether a geography-related upgrade is necessary by executing the following query:
 

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```
  - b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\<<database type>>.  
  
If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.
19. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
20. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.



- d. Click OK.
- e. Choose Actions, and then click Delete Contents.
- f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 8.8.2 Upgrading Oracle BI Applications From Version 7.9.6.2 to 7.9.6.3

Follow this procedure to upgrade from Oracle BI Applications version 7.9.6.2 to version 7.9.6.3 if your data warehouse database platform is Teradata.

### To upgrade the data warehouse schema and migrate data for a Teradata database

1. Execute the following script:
  - 7963\_UPGRADE\_TMP\_TABLES.sql
  - a. Open the SQL client for your database type.
  - b. Navigate to the folder OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  - c. Open the <version>\_UPGRADE\_TMP\_TABLES.sql file, and copy the contents into the SQL client.
  - d. Execute the script.
2. The teradata.missing.defaults.properties file contains global default column values for data warehouse tables with NOT NULL columns for which default values are not specified in DAC. You can change the default values specified in the teradata.missing.defaults.properties file to match your environment. The file is located in \bifoundation\dac\conf\sqlgen\sql\teradata.
3. Generate upgrade scripts from DAC.
  - a. Log into DAC.
  - b. From the menu bar, select Tools, then ETL Management, and then Configure.
  - c. In the Sources dialog box, select Teradata as the target data warehouse database platform, and select the appropriate source data warehouse database platform.  
The Configuration tab of the Data Warehouse Configuration Wizard is active.
  - d. Select "Generate upgrade statements for Data Warehouse Tables," and click Next.  
The Data Warehouse SQL tab is active.
  - e. Do the following:
    - Select the appropriate physical data source name for the Teradata database.
    - In the container field, enter a custom container name for which you want to generate upgrade scripts. Alternatively, leave the field blank to generate upgrade scripts for all containers.

The Data Warehouse Configuration Wizard generates the following files and stores them in \bifoundation\DAC\conf\sqlgen\sql\teradata:

**upgrade-regular.sql.** This file contains scripts that DAC has verified are necessary to upgrade the data warehouse schema. The scripts handle new or altered tables and columns. For example, they may increase the size of simple data types or change a column from null to not null.

**upgrade-questionable.sql.** This file contains scripts that DAC suggests may be required to upgrade the data warehouse schema. These scripts are intended to be an aid in determining upgrade requirements and must be reviewed by a Teradata DBA and corrected if necessary.

**upgradedwtables\_sql.log.** This file is the log file that corresponds to the upgrade-regular.sql file. It is for information purposes only.

**upgrade-issues.log.** This file is the log file that corresponds to the file upgrade-questionable.sql. It is for information purposes only.

- f. Open the upgrade-regular.sql and upgrade-questionable.sql files, and search for the following statement to locate columns that have been added or changed.
 

```
ALTER TABLE <Table Name> ADD <Column Name><attributes>;
```
  - g. For all columns with the ADD command, check to see whether any columns exist already, and if they do, check whether they have STATISTICS defined on them.
  - h. For all columns that have STATISTICS defined on them, back up the column definition and then drop the STATISTICS.
  - i. Execute the scripts in the upgrade-regular.sql file.
  - j. Review the scripts in the upgrade-questionable.sql file. Edit the scripts as needed, and execute them.
  - k. Recreate STATISTICS on any column on which they were dropped.
4. Execute the following scripts:
    - 7963\_UPDATE\_DEFAULTS.sql
  5. Execute the 7963\_Backup\_Tables.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\Teradata.
  6. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.
    - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.5, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle."](#)
    - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.6, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server."](#)
  7. Configure the Informatica parameter file 7963\_UPG\_PARAMS.txt. For instructions, see [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt."](#)
  8. If you are upgrading Oracle Financial Analytics, set the parameters as specified in [Appendix A.3.5.1, "Setting Parameters for All Source Systems."](#)
  9. If you are using JD Edwards 8.11 or 8.12, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE\_811SP1\_812 and execute the UPGRADE\_FACTS workflow.
 

If you are using JD Edwards 9, in Informatica Workflow Manager, navigate to the folder UPGRADE\_7962\_to\_7963\_JDE90 and execute the UPGRADE\_FACTS workflow.
  10. Upgrade geography-related attributes if necessary for your environment:

- a. Determine whether a geography-related upgrade is necessary by executing the following query:

```
SELECT COUNT(1) as row_count, '0' as X_CUSTOM FROM W_GEO_D_7963 WHERE
COUNTRY LIKE '% 'OR COUNTRY LIKE ' %'OR STATE_PROV LIKE '% 'OR STATE_PROV
LIKE ' %'OR CITY LIKE '% 'OR CITY LIKE ' %'OR ZIPCODE LIKE '% 'OR ZIPCODE
LIKE ' %'
```

- b. If the query returns a count greater than zero, run the 7963\_UPGRADE\_POST\_SCRIPT.sql script. This file is located in OracleBI\dwrep\Upgrade\DbScripts\.

If the query returns a count of zero, you do not need to run the 7963\_UPGRADE\_POST\_SCRIPT.sql script.

11. Use the reset\_infa\_seq\_gen.bat script to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).
  - If your Informatica Repository runs on an Oracle database, for instructions, see [Appendix D.2, "Running reset\\_infa\\_seq\\_gen\\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle."](#)
  - If your Informatica Repository runs on an MSSQL Server database, for instructions, see [Appendix D.3, "Running reset\\_infa\\_seq\\_gen\\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server."](#)
12. Locate the 7962\_RESET\_REFRESH\_DATES.txt file in the folder OracleBI\dwrep\Upgrade\Informatica\SrcFiles.

In the next step, you will reset the refresh dates to NULL for the tables listed in 7962\_RESET\_REFRESH\_DATES.txt.

13. In DAC, do the following:
  - a. Navigate to the Setup view, and click the Physical Data Sources tab.
  - b. In the top pane list, select DataWarehouse. (If you customized the name of the connection for the data warehouse, select the appropriate connection for the data warehouse database.)
  - c. Click the Refresh Dates subtab (in the lower pane).
  - d. Query for the tables listed in the 7962\_RESET\_REFRESH\_DATES.txt file, and, for each table, set the value in the Refresh Date column to NULL
14. Once the data migration steps above are complete, you can delete the Upgrade repository to avoid any accidental use or confusion.
  - a. In the Informatica Administrator, select the Upgrade repository service.
  - b. In the General Properties area of the Properties tab, click Edit.
  - c. Make sure the operating mode of the repository service is set to Exclusive.
  - d. Click OK.
  - e. Choose Actions, and then click Delete Contents.
  - f. In the Delete contents for <repository name> dialog, enter the repository username and password, and then click OK.

## 8.9 Localizing the Policy Store

The Oracle BI Applications 7.9.6.3 installer contains different policy stores for different languages. The default policy store that is deployed during the installation is in English. If you want to configure the policy store for any of the other available languages, perform the procedure below.

### To configure the policy store for languages other than English

1. Stop the Oracle BI Services.
2. Copy the system-jazn-data\_<LN>.xml file from \$ORACLE\_BI\_HOME\biapps\admin\provisioning\localization\ to \$DOMAIN\_HOME\config\fmwconfig\  
  
For example, to implement the policy store for French, you would copy the file system-jazn-data\_fr.xml into the \fmwconfig directory.
3. Back up the existing system-jazn-data.xml file in \$DOMAIN\_HOME\config\fmwconfig\  
  
For example, to implement the policy store for French, you would copy the file system-jazn-data\_fr.xml into the \fmwconfig directory.
4. Rename \$DOMAIN\_HOME\config\fmwconfig\system-jazn-data\_<LN>.xml to system-jazn-data.xml.
5. Start Oracle BI Services.

## 8.10 Importing the Oracle BI Applications Version 7.9.6.3 LDIF File Into the Embedded LDAP Server

In this procedure, you will import the standard (out-of-the-box) Oracle BI Applications version 7.9.6.3 LDIF file into the WebLogic Server embedded LDAP server.

### To import the Oracle BI Applications version 7.9.6.3 LDIF file

1. Log in to the WebLogic Server Administration Console using the URL `http://<host name>:7001/console`.
2. Select the name of the security realm into which the LDIF file is to be imported (for example, myrealm).
3. Select **Providers** and then the type of provider into which the LDIF file is to be imported (for example, **Providers**, and then **Authentication**).
4. Select the security provider in which the LDIF file is to be imported and select **Migration**, and then select **Import**.
5. Click **Save**.

## 8.11 Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version

Use the Oracle BI Enterprise Edition Upgrade Assistant to upgrade the following repositories and presentation catalog from the Oracle BI Enterprise Edition 10g format to the 11g format:

- The standard (out-of-the-box) Oracle BI Repository. This repository is the one that you originally received from Oracle as part of your current Oracle BI Applications installation (the version from which you are upgrading). Standard repositories from previous releases are available in the installation folder \$ORACLE\_BI\_HOME\biapps\upgrade.

- The Oracle BI Repository you are currently using (the repository deployed in production).
- The Presentation Catalog you are currently using (the catalog deployed in production).

For instructions on using the Upgrade Assistant, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*.

---

**Note:** Your current policy store will be merged with the Oracle BI Applications release 7.9.6.3 policy store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final policy store will be the super set of Oracle BI Applications release 7.9.6.3 policy store and the custom and current policy information from the 10g repository.

Also, your current users and groups from the repository will be merged with the Oracle BI Applications release 7.9.6.3 identity store while upgrading the Oracle BI repository from the 10g format to the 11g format. The final identity store will be the super set of the Oracle BI Applications release 7.9.6.3 identity store and the identity store information (users and groups) from the 10g repository.

If you require the policy store to be in a language other than English, perform the steps in [Section 8.9, "Localizing the Policy Store."](#)

---

## 8.12 Merging Oracle BI Applications Version 7.9.6.3 Repository With Customer Repository

---

**Note:** Before you begin this procedure, you must have already upgraded the Oracle BI repository you are currently using and the standard (out-of-the-box) Oracle BI repository version 7.9.x (the version from which you are upgrading) to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

This process merges your customizations of a prior release of the Oracle BI repository with the new release of the Oracle BI repository. Before you begin this process, make sure you have backed up and renamed your existing repository.

To upgrade the repository, perform the following tasks:

- [Section 8.12.1, "Preparing for the Oracle BI Repository Upgrade"](#)
- [Section 8.12.2, "Equalizing the Oracle BI Repositories"](#)
- [Section 8.12.3, "Comparing the Oracle BI Repositories"](#)
- [Section 8.12.4, "Merging the Oracle BI Repositories"](#)

The tasks in this section refer to multiple releases of the Oracle BI repository. [Table 8-1](#) provides the names and descriptions of the repositories used in the examples in this section.

**Table 8–1 Names of Analytics Repositories used in Examples**

| Name of Repository                 | Description                                                                                                                                                                                             |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OracleBIAnalyticsApps_79x.rpd      | The standard Oracle BI repository for the version you are upgrading from.<br><br><b>Note:</b> Standard repositories from previous releases are available in the folder \$ORACLE_BI_HOME\biapps\upgrade. |
| OracleBIAnalyticsApps.rpd          | The standard Oracle BI repository for the version you are upgrading to.                                                                                                                                 |
| Customer_OracleBIAnalyticsApps.rpd | The Oracle BI repository that contains your customizations for the version you are upgrading from.                                                                                                      |
| Merged_Repository_OracleBI.rpd     | The Oracle BI repository that contains your customizations for the version you are upgrading to.                                                                                                        |

### 8.12.1 Preparing for the Oracle BI Repository Upgrade

Follow this procedure to prepare for the repository upgrade.

#### To prepare for the Analytics repository upgrade

- Set up a directory for the merge process, such as \OracleBIUpgrade, and create the following subfolders:
  - Original
  - AfterEqualize
  - AfterMerge
  - AfterManualWork
  - AfterRegressions
- Copy the following repositories to the folder \OracleBIUpgrade\Original:
  - The standard repository that you upgraded to the 11g format in [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version"](#) (for example, OracleBIAnalyticsApps\_79x.rpd).
  - The production repository that you also upgraded to the 11g format in [Section 8.11](#) (for example, Customer\_OracleBIAnalyticsApps.rpd).
  - The Oracle BI repository from the latest installation (for example, OracleBIAnalyticsApps.rpd)

If, in your current environment, you are running Oracle BI Applications for one or more modules using a Oracle BI repository in which you extracted the corresponding projects for the modules from the standard Oracle BI repository file you received from the previous release, you need to extract the same projects from the OracleBIAnalyticsApps\_79x.rpd file and use this as your original repository. (If you have the original repository that you extracted during the last upgrade, you can use it as the original repository file.) This will prevent you from losing any new metadata you would like to add in this upgrade.

Also, if you customized the Oracle BI repository by trimming a large number of objects and you would like to get those objects back during the current upgrade, you need to trim the OracleBIAnalyticsApps\_79x.rpd file in the same way and use the modified version as the original repository file. This will prevent you from losing any new metadata you would like to add in this upgrade.

## 8.12.2 Equalizing the Oracle BI Repositories

The Merge feature in the Administration Tool relies on a change detection algorithm to determine the changes that need to be made to upgrade repositories correctly. For the algorithm to work correctly, it has to determine which objects in the three repositories (for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) are equivalent.

The point of this step is to determine for every object in the OracleBIAnalyticsApps.rpd and the Customer\_OracleBIAnalyticsApps.rpd whether it is coming from the OracleBIAnalyticsApps\_79x.rpd.

Equivalence between objects is established using the Administration Tool's Equalize feature. The file that you specify in the Output option (-O) is the only file that is modified during the equalization process.

The Equalize feature has several mechanisms for determining whether an object in two different repositories is semantically the same:

- **Fully Qualified Name.** If an object in one repository has the same fully qualified name as another object of the same class in another repository, then the two objects are declared equal.
- **Simple String Substitution.** Equivalence can be declared between two objects of the same class in two repositories whose only difference is that some key characters in their names differ. The equalizerpds executable file ignores those characters while checking fully qualified names. For example, "Core"."W\_DAY\_D" might be considered equivalent to "Core"."W DAY D" if the characters "\_" and " " have been declared as equivalent.
- **Rename File.** When none of the preceding rules are applicable, equivalence can be manually declared using a script as input to the equalizerpds executable file. Oracle ships the rename files (MAP) for the major releases. The files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder. You can also create your own rename files for customizations not covered in the files that Oracle ships. You can open and edit the rename files in Microsoft Excel.

Table 8–2 provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

**Table 8–2 Rename MAP Files to Be Used for Various Releases**

| Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version) | Rename MAP File to Be Used |
|----------------------------------------------------------------------------------------------------------|----------------------------|
| Siebel Business Analytics Applications 7.0.x                                                             | Not available              |
| Siebel Business Analytics Applications 7.5.x                                                             | Not available              |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)                            | Rename77-7963.map          |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)                                | Rename771-7963.map         |
| Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release                  | Rename782-7963.map         |
| Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release                   | Rename783-7963.map         |
| Oracle BI Applications 7.9.0                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.1                                                                             | Rename79x-7963.map         |

**Table 8–2 (Cont.) Rename MAP Files to Be Used for Various Releases**

| <b>Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version)</b> | <b>Rename MAP File to Be Used</b> |
|-----------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Oracle BI Applications 7.9.2                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.3                                                                                    | Rename793to7963.map               |
| Oracle BI Applications 7.9.4                                                                                    | Rename794to7963.map               |
| Oracle BI Applications 7.9.5                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.5.1                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.5.2                                                                                  | Rename7951to7963.map              |
| Oracle BI Applications 7.9.6                                                                                    | Rename79x-7963.map                |
| Oracle BI Applications 7.9.6.2                                                                                  | Not required                      |

The syntax of the equalizerpds command is as follows:

```
equalizerpds.exe [-B [password1]] -C base_repository_name [-E [password2]] -F
repository2_name [-J udm1_utf8_file_name_equalization] [-O output_repository_name]
[-Y equalStringSet]
-Y Treat the characters as equals.
/? Display this usage information and exit.
```

### To equalize a repository

1. Copy the appropriate MAP file from the \$ORACLE\_BI\_HOME\biapps\upgrade folder into the folder where you will execute equalizerpds.exe, for example, \OracleBIUpgrade\Original.
2. Run equalizerpds.exe to equalize the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsApps\_793.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-D Administrator -E SADMIN
-F \\OracleBIUpgrade\Original\OracleBIAnalyticsApps.rpd
-O \\OracleBIAnalyticsUpgrade\AfterEqualize\OracleBIAnalyticsApps.rpd
-X -J rename7x-79.map
```

The MAP files are located in the \$ORACLE\_BI\_HOME\biapps\upgrade folder.

If the equalizerpds.exe executable file runs correctly, no errors are returned.

3. Run equalizerpds.exe to equalize your customized repository (for example, Customer\_OracleBIAnalyticsApps.rpd) with the original repository (for example, OracleBIAnalyticsApps\_79x.rpd). An example of the equalizerpds command is as follows:

```
equalizerpds -B SADMIN
-C \\OracleBIUpgrade\Original\OracleBIAnalyticsApps_791.rpd
-E SADMIN
-F \\OracleBIUpgrade\Original\Customer_OracleBIAnalyticsApps.rpd
-O \\OracleBIUpgrade\AfterEqualize\Customer_OracleBIAnalyticsApps.rpd
```

The execution of equalizerpds that equalizes the customer repository with the original repository does not use the rename file.



Make sure that the original repository is copied unchanged into its new location so that after running the script, all three repositories are contained within the \OracleBIUpgrade\AfterEqualize directory.

4. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

### 8.12.3 Comparing the Oracle BI Repositories

Follow this procedure to compare your existing repository with the new version to which you are upgrading.

#### To compare the Oracle BI repositories

- Use the Administration Tool's Compare Repositories feature to analyze the differences between your existing repository and the new version of the repository to which you are upgrading. Note where elements have been created, removed, or changed in the new version. Consider whether you can use the new metadata and retire customizations you made in the existing repository.

For instructions on how to use the Administration Tool's Compare Repositories feature, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

### 8.12.4 Merging the Oracle BI Repositories

In this procedure, you execute the main algorithm to upgrade the repository. For more information on merging the repositories, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

#### To merge versions of the Oracle BI repositories

1. Copy the three repositories (for example, OracleBIAnalyticsApps\_79x.rpd, OracleBIAnalyticsApps.rpd, and Customer\_OracleBIAnalyticsApps.rpd) to the AfterMerge folder.
2. Open the repository from the latest installation (for example, OracleBIAnalyticsApps.rpd) in the \OracleBIUpgrade\AfterMerge folder.
3. From the Administration Tool menu bar, select **File**, then select **Merge**.
4. Select the Merge Type option **Full Repository Merge**.
5. In the Select Original Repository dialog box, select the original repository (for example, OracleBIAnalyticsApps\_79x.rpd).
6. Enter the password for the original repository.
7. In the **Select Modified Repository** dialog box, select the repository that contains the customizations you made to the previous version of the Analytics repository.
8. Enter the password for the modified repository.
9. In the **Saved Merged Repository as** dialog box, enter the name for the merged repository, for example, Merged\_Repository\_OracleBIAnalyticsApps.rpd.  
The new repository will contain the final results of the merge.
10. Select the **Equalize during merge** check box.
11. Click **Next**.

12. In the Decision drop-down list, select the action you want to take regarding the repository change, or accept the default action.
13. To locate subsequent rows with empty Decision fields, click the Decision header cell.

When all rows have a value in the Decision field, the Finish button is enabled.

14. Select the **Check consistency of the merged RPD** check box.
15. Click **Finish**.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

16. The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file if the object is in the Current repository, or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

17. Copy the repository to the folder \OracleBIUpgrade\AfterManualWork.

## 8.13 Merging Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

---

You need to perform the following tasks to complete the presentation merge process.

- [Section 8.13.1, "Trimming the Input Presentation Catalog"](#)
- [Section 8.13.2, "Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current \(Custom\) Catalog"](#)
- [Section 8.13.3, "Testing the Results of the Presentation Catalog Upgrade"](#)

These procedures use the following terminology to identify the various presentation catalogs you will use in the merge process:

- The *original* presentation catalog. This catalog is the output from the step you performed in [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the version of the unmodified presentation catalog you received with the Oracle BI Applications release you are upgrading from that you upgraded to the 11g format.
- The *new* presentation catalog. This is the standard Oracle BI Applications 7.9.6.3 presentation catalog that is installed with Oracle BI Applications.
- The *current* presentation catalog. This catalog is the output from the step you performed in [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#) It is the presentation catalog currently in use at your organization that you upgraded to the 11g format.

---

**Caution:** In releases of Oracle BI Applications previous to 7.9, the Presentation Catalog (formerly known as the Siebel Analytics Web Catalog) was stored in a single file rather than in a directory structure of individual files. If you have a previous version of the Presentation Catalog, you will need to convert it to the new format. For more information about how to convert the Presentation Catalog to the new format, see the *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*

---

### 8.13.1 Trimming the Input Presentation Catalog

Before you upgrade and merge your current Presentation Catalog with the new Presentation Catalog, determine which of the existing content you want to keep and which new content you want to incorporate. Review your existing Presentation Catalog and determine the usage patterns of reports and dashboards. Note that some of the preconfigured content in the existing catalog may appear in the new version in a redesigned format. In addition, the new version includes completely new content. After you have decided the content that is to make up your enterprise Presentation Catalog, trim the input catalogs using the Catalog Manager. For information on trimming catalogs, see *Oracle Business Intelligence Presentation Services Administration Guide*.

### 8.13.2 Merging the Oracle BI Applications Version 7.9.6.3 Presentation Catalog With Current (Custom) Catalog

---

**Note:** Before you begin this procedure, you must have already upgraded your current (custom) catalog to the Oracle BI Enterprise Edition 11g format. For more information, see [Section 8.11, "Upgrading the Oracle BI Repository and Presentation Catalog to the Oracle BI Enterprise Edition 11g Version."](#)

---

Follow the instructions in this section to merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the catalog you are currently using. This procedure uses the Oracle BI Enterprise Edition Catalog Manager. High-level steps for the merge process are provided in this section. For detailed instructions on using the Catalog Manager, see Chapter 17, "Configuring and Managing the Oracle BI Presentation Catalog," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

**To merge the Oracle BI Applications version 7.9.6.3 presentation catalog with the current (custom) catalog**

1. Start the Catalog Manager, and open your current (custom) presentation catalog in offline mode.

For instructions, see Section 17.4, "Starting Catalog Manager and Opening Catalogs," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

2. Start another instance of the Catalog Manager, and open the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog in offline mode.
3. Copy and paste the custom objects from your current presentation catalog into the Oracle BI Applications version 7.9.6.3 (out-of-the-box) presentation catalog.

For instructions, see Section 17.6.1, "Working with Objects in Catalog Manager," in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

4. If you created custom catalog groups that are not in the version 7.9.6.3 presentation catalog, you need to convert the custom catalog groups to application roles in order to maintain consistency with the version 7.9.6.3 presentation catalog.

To convert custom catalog groups to application roles, run the following command:

```
runcat.cmd/runcat.sh -cmd replaceAccountInPermissions -old <Catalog Group Name> -oldType group -new <App Role Name> -newType role -offline <catalog path>
```

For example:

```
runcat.cmd -cmd replaceAccountInPermissions -old "AP Analyst" -oldType group -new "AP Analyst" -newType role -offline c:/SampleWebcat
```

This command replaces a specified account with another in all catalog object ACLs and privileges in the presentation catalog, entirely in the offline mode. If an entirely new application role is specified as the replacement account, then it is necessary to refresh the GUIDs in the presentation catalog before it can be used.

**Note:** If the specified replacement user or group is not already present in the presentation catalog already, then this operation will fail.

For more information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition 11g Release 1 (11.1.1)*.

### 8.13.3 Testing the Results of the Presentation Catalog Upgrade

---

---

**Note:** Before you perform this step, you must first migrate the data into the upgraded data warehouse.

---

---

The presentation catalog upgrade functionality does not automatically carry over object permissions; therefore, you should review the presentation catalog object permissions before you perform this step.

This step ensures that the upgraded reports and the new preconfigured reports are functional and render correct results within the new, merged presentation catalog. This step is typically performed by visually inspecting the final results of the complete end-to-end upgrade process.

For upgraded reports, the preferred approach for comparison purposes is to have side-by-side environments, and have users review specific dashboard content between the two environments. Examine not only the look and feel of the application but also the data contained in the reports to make sure the content remains the same. It is recommended that you request users to use various elements of the user interface to validate results, such as global prompts, column selectors, report filters, drills, and navigations, as they normally do on a day-to-day basis.

Also review the overall visibility and administrative settings in the new presentation catalog to ensure they are correct. Pay careful attention to the visibility rules that are established for any content that was migrated during the upgrade. You might have to manually adjust these settings.

## 8.14 Regression Testing the Oracle BI Repository Merge

In performing a regression test for the repository merge, the objective is to collect a set of logical SQL statements that are used for reports and to verify that they continue to work with the new metadata. For this purpose, it is recommended that you perform the following procedure.

### To perform regression testing

1. Run the reports that are necessary to include in the regression suite. These reports might be a subset of the reports in the Presentation Catalog.
2. Collect the logical SQL generated in the previous step. You can do this using Usage Tracking or by parsing the query log file.

For information about Usage Tracking, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

3. Execute the logical SQL against the old repository using the command line utility nQCmd.exe located in \$ORACLE\_BI\_HOME\bifoundation\server\bin, and save the results to a file.

For information about the nQCmd.exe utility, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Edit the logical SQL test scripts to account for the name changes or modifications resulting from the upgrade.
5. Execute the edited logical SQL against the merged repository, and save the results.
6. Compare the results from the steps above and try to explain the differences. If it is determined that these differences are due to the upgrade process, then you have to correct them manually.

This repository now contains the merged content from the new OracleBIAnalyticsApps.rpd and the production repository.



# Part V

---

## Appendices

Part IV contains the following appendixes:

- [Appendix A, "Setting Parameters in Informatica Parameter Files"](#)
- [Appendix B, "Generating and Running Scripts to Drop and Create Indexes"](#)
- [Appendix C, "Creating Indexes for Oracle Financial Analytics"](#)
- [Appendix D, "Setting Database-Specific Parameters for the reset\\_infa\\_seq\\_gen.bat File"](#)
- [Appendix E, "Using the DDLimp Utility"](#)
- [Appendix F, "Upgrading the Oracle BI Repository for Industry-Specific Analytics Applications"](#)





---

---

# Setting Parameters in Informatica Parameter Files

This appendix provides instructions for setting parameters in the Informatica parameter files. It contains the following topics:

- [Section A.1, "Setting Common Parameters in Informatica Parameter Files"](#)
- [Section A.2, "Setting Source System-Specific Parameters in Informatica Parameter Files"](#)
- [Section A.3, "Setting Application-Specific Parameters in Informatica Parameter Files"](#)

## A.1 Setting Common Parameters in Informatica Parameter Files

This section provides instructions for configuring the <Oracle BI Applications version>\_UPG\_PARAMS.txt file. You need to configure these files for each version of Oracle BI Applications to which you are upgrading.

This section contains the following topics:

- [Section A.1.1, "Setting Parameters in 793\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.2, "Setting Parameters in 794\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.3, "Setting Parameters in 795\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.4, "Setting Parameters in 7951\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.5, "Setting Parameters in 796\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.6, "Setting Parameters in 7961\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt"](#)
- [Section A.1.8, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt"](#)

### A.1.1 Setting Parameters in 793\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 793\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.3.

#### To set parameters in the 793\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 793\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.

2. Set the parameter \$\$ETL\_PROC\_WID to the latest ETL\_PROC\_WID value from the database. You can get this value from W\_PARAM\_G.ETL\_PROC\_WID.
3. Set the parameter \$\$DATASOURCE\_NUM\_ID to the relevant value from the source system setup.
4. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For SQL Server databases, use "CONVERT(datetime, '1899-01-01')"
  - For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"

### A.1.2 Setting Parameters in 794\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 794\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.4.

#### To set parameters in the 794\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 794\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Set the parameter \$\$ETL\_PROC\_WID to the latest ETL\_PROC\_WID value from the database. You can get this value from W\_PARAM\_G.ETL\_PROC\_WID.
3. Set the parameter \$\$DATASOURCE\_NUM\_ID to the relevant value from the source system setup.
4. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For SQL Server databases, use "CONVERT(datetime, '1899-01-01')"
  - For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"

### A.1.3 Setting Parameters in 795\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 795\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.5.

#### To set parameters in the 795\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 795\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.

2. Set the parameter \$\$ETL\_PROC\_WID to the latest ETL\_PROC\_WID value from the database. You can get this value from W\_PARAM\_G.ETL\_PROC\_WID.
3. Set the parameter \$\$DATASOURCE\_NUM\_ID to the relevant value from the source system setup.
4. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For SQL Server databases, use "CONVERT(datetime, '1899-01-01')"
  - For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"

#### A.1.4 Setting Parameters in 7951\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 7951\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.5.1.

##### To set parameters in the 7951\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 7951\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Set the parameter \$\$ETL\_PROC\_WID to the latest ETL\_PROC\_WID value from the database. You can get this value from W\_PARAM\_G.ETL\_PROC\_WID.
3. Set the parameter \$\$DATASOURCE\_NUM\_ID to the relevant value from the source system setup.
4. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For SQL Server databases, use "CONVERT(datetime, '1899-01-01')"
  - For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"

#### A.1.5 Setting Parameters in 796\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 796\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.6.

##### To set parameters in the 796\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 796\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.

2. In the 796\_UPG\_PARAMS.txt file, set the following parameters:
  - a. `$$ETL_PROC_WID`. Set this parameter to the relevant value from the source system setup. You can get this value from `W_PARAM_G.ETL_PROC_WID`
  - b. `$$DATASOURCE_NUM_ID`. Set this parameter to the relevant value from the source system setup.
  - c. `$$INITIAL_EXTRACT_DATE`. Set this parameter to the initial extraction data of the data warehouse.
  - d. `$$WH_DATASOURCE_NUM_ID`. Set this parameter to the data source number ID you have set up for the data warehouse. This value should be the same data source number ID in the warehouse table `W_DAY_D`. If this is not set up correctly, the upgrade maps will fail with a `NULL ROW_WID`. Make sure this parameter is setup appropriately for your environment.
  - e. `$$START_DATE`. Get this value from the Source System Parameters tab in DAC.
  - f. `$$END_DATE`. Get this value from the Source System Parameters tab in DAC.
  - g. `$$MASTER_ORG`. Get this value from the Source System Parameters tab in DAC.
  - h. `$$INV_PROD_CAT_SET_ID1`. Get this value from the Source System Parameters tab in DAC.
  - i. `$$PROD_CAT_SET_ID1`. Get this value from the Source System Parameters tab in DAC.
  - j. Set the parameter `$$IS_SOURCE_PRE_80` to Y if your source OLTP application was on a version prior to Siebel 8.0 before you began the upgrade process. Otherwise, set this parameter to N.
  - k. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
    - For Oracle databases, use `"TO_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"`
    - For SQL Server databases, use `"CONVERT(datetime, '1899-01-01')"`
    - For DB2 databases, use `"TO_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"`
    - For Teradata databases, use `"cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"`

### A.1.6 Setting Parameters in 7961\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 7961\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.6.1.

#### To set parameters in the 7961\_UPG\_PARAMS.txt file

1. Navigate to the folder `OracleBI\dwrep\Upgrade\Informatica\ParameterFiles` and copy the file `7961_UPG_PARAMS.txt` into the `SrcFiles` folder on the Informatica Server machine, for example, `<INFA_HOME>\server\infa_shared\SrcFiles`.
2. In the `7961_UPG_PARAMS.txt` file, set the following parameters:

- a. `$$ETL_PROC_WID`. Set this parameter to the relevant value from the source system setup. You can get this value from `W_PARAM_G.ETL_PROC_WID`
- b. `$$DATASOURCE_NUM_ID`. Set this parameter to the relevant value from the source system setup.
- c. `$$INITIAL_EXTRACT_DATE`. Set this parameter to the initial extraction data of the data warehouse.
- d. `$$WH_DATASOURCE_NUM_ID`. Set this parameter to the data source number ID you have set up for the data warehouse.
- e. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use `"TO_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"`
  - For SQL Server databases, use `"CONVERT(datetime, '1899-01-01')"`
  - For DB2 databases, use `"TO_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"`
  - For Teradata databases, use `"cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"`

### A.1.7 Setting Parameters in 7962\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 7962\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.6.2.

**Note:** If your source system is PeopleSoft, you must also set the parameters in [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems."](#)

#### To set parameters in the 7962\_UPG\_PARAMS.txt file

1. Navigate to the folder `OracleBI\dwrep\Upgrade\Informatica\ParameterFiles` and copy the file `7962_UPG_PARAMS.txt` into the `SrcFiles` folder on the Informatica Server machine, for example, `<INFA_HOME>\server\infa_shared\SrcFiles`.
2. In the `7962_UPG_PARAMS.txt` file, set the following parameters:
  - a. `$$ETL_PROC_WID`. Set this parameter to the relevant value from the source system setup. You can get this value from `W_PARAM_G.ETL_PROC_WID`
  - b. `$$DATASOURCE_NUM_ID`. Set this parameter to the relevant value from the source system setup.
  - c. `$$INITIAL_EXTRACT_DATE`. Set this parameter to the initial extraction data of the data warehouse.
  - d. `$$WH_DATASOURCE_NUM_ID`. Set this parameter to the data source number ID you have set up for the data warehouse.
  - e. Search for parameter values defined with the "TO\_DATE" function. Edit the function to use the appropriate function for data conversion based on the database type:
    - For Oracle databases, use `"TO_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"`
    - For SQL Server databases, use `"CONVERT(datetime, '1899-01-01')"`

- For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
- For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"

### A.1.8 Setting Parameters in 7963\_UPG\_PARAMS.txt

Follow this procedure to set parameters in the 7963\_UPG\_PARAMS.txt file. This procedure is applicable for upgrades to Oracle BI Applications release 7.9.6.3.

#### To set parameters in the 7963\_UPG\_PARAMS.txt file

1. Navigate to the folder OracleBI\dwrep\Upgrade\Informatica\ParameterFiles and copy the file 7963\_UPG\_PARAMS.txt into the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. In the 7963\_UPG\_PARAMS.txt file, set the following global parameters:
  - a. **\$\$ETL\_PROC\_WID**. Set this parameter to the relevant value from the source system setup. You can get this value from W\_PARAM\_G.ETL\_PROC\_WID
  - b. **\$\$WH\_DATASOURCE\_NUM\_ID**. Set this parameter to the data source number ID you have set up for the data warehouse.
3. In the 7963\_UPG\_PARAMS.txt file, set the following common dimension parameters:
  - a. Locate the section in the file that lists common dimension parameters for your source system, and set the following parameters.
  - b. **\$\$DATASOURCE\_NUM\_ID**. Set this parameter to the relevant value from the source system setup.
  - c. **\$\$INITIAL\_EXTRACT\_DATE**. Set this parameter to the initial extraction data of the data warehouse.
  - d. **\$\$DFLT\_LANG**. The default value for this parameter is \$\$DFLT\_LANG=US. This value is appropriate for Oracle source systems when the transactional database language is English (US).

For Siebel and JD Edwards source systems with the transactional database language as English (US), set the value for the parameter to \$\$DFLT\_LANG=ENU.

If the default language of your transactional database is not English (US), you need to set the DFLT\_LANG parameter to the appropriate language for your data source database. To find the value to specify, execute the following query against the transactional database.

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

**Note:** PeopleSoft source systems do not use the \$\$DFLT\_LANG parameter.

## A.2 Setting Source System-Specific Parameters in Informatica Parameter Files

This section provides instructions for setting parameters in the Informatica parameter files that are specific to various source systems. You may need to set or update these parameters depending on your environment. The topic headings below indicate the source system and version of the parameter file that require configuration.

This section includes the following topics:

- [Section A.2.1, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle EBS 11i Source Systems"](#)
- [Section A.2.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Siebel Source Systems"](#)
- [Section A.2.3, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt for PeopleSoft Source Systems"](#)
- [Section A.2.4, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Siebel Industry Applications Source Systems"](#)

**Note:** If you are using Oracle Financial Analytics and your source system is either PeopleSoft or Oracle EBS 11.5.10 family pack OIE.I and OIE.J, there are additional parameters you must set. For instructions, see [Section A.3.5, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Financial Analytics."](#)

### A.2.1 Setting Parameters in 796\_UPG\_PARAMS.txt for Oracle EBS 11i Source Systems

This procedure is only applicable to Oracle EBS 11i source systems and for upgrades to Oracle BI Applications release 7.9.6.

**To set parameters in the 796\_UPG\_PARAMS.txt file for Oracle EBS 11i source systems**

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the file 796\_UPG\_PARAMS.txt.
3. Note the values for the following parameters:
  - \$\$ORA\_DATASOURCE\_NUM\_ID\_LIST
  - \$\$GRAIN
  - \$\$GBL\_DATASOURCE\_NUM\_ID
  - \$\$QUALIFICATION\_CATEGORY\_LIST
4. In DAC, go to the Design view, and select the appropriate custom container.
5. Select the Source System Parameters tab.
6. Query for the parameters listed in Step 3, and compare the values.
7. If necessary, change the values for the parameters in the 796\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
8. Save the 796\_UPG\_PARAMS.txt file.

### A.2.2 Setting Parameters in 796\_UPG\_PARAMS.txt for Siebel Source Systems

This procedure is only applicable to Siebel source systems and for upgrades to Oracle BI Applications release 7.9.6.

**To set parameters in the 796\_UPG\_PARAMS.txt file for Siebel source systems**

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the file 796\_UPG\_PARAMS.txt.

3. Note the values for the following parameters:
  - \$\$HI\_DT
  - \$\$LOW\_DT
  - \$\$NAME\_ORDER\_WITH\_FIRSTNAME
4. In DAC, go the Design view, and select the appropriate custom container.
5. Select the Source System Parameters tab.
6. Query for the parameters listed in Step 3, and compare the values.
7. If necessary, change the values for the parameters in the 796\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
8. Save the 796\_UPG\_PARAMS.txt file.

### A.2.3 Setting Parameters in 7962\_UPG\_PARAMS.txt for PeopleSoft Source Systems

If your source system is PeopleSoft, you must follow the instructions in this section after completing the instructions in [Section A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt."](#) This procedure is applicable for upgrades to Oracle BI Applications release 7.9.6.2.

#### To set parameters in the 7962\_UPG\_PARAMS.txt file for PeopleSoft source systems

1. Open the 7962\_UPG\_PARAMS.txt file.
 

**Note:** Make sure you perform the procedure in [Section A.1.7, "Setting Parameters in 7962\\_UPG\\_PARAMS.txt"](#) before you begin this procedure.
2. If you are using PeopleSoft version 8.9 or higher, set the following parameters in the 7962\_UPG\_PARAMS.txt file specific to multiple product category enhancement:
  - a. For the \$\$TREE\_NAME1 parameter, set the value to a tree name for which you want to include the product categories. Generally, this value is ALL\_PURCHASE\_ITEMS. You can use the following SQL to obtain the TREE\_NAMES:
 

```
SELECT * FROM PSTREEDEFN WHERE TREE_STRCT_ID = 'ITEMS'
```
  - b. The TREE\_STRUCT in the PeopleSoft Tree that defines the Item Category Tree generally has the value ITEMS. If you changed this value, you must update the parameter \$\$TREE\_STRUCT\_ID\_LIST with the appropriate value.
3. Verify the value for the PSFT\_OLTP\_VER parameter matches the version of PeopleSoft you are using.

| PeopleSoft Version     | Enter Parameter Value |
|------------------------|-----------------------|
| PeopleSoft version 8.9 | PSFT_OLTP_VER=90      |
| PeopleSoft version 9.0 | PSFT_OLTP_VER=90      |
| PeopleSoft version 9.1 | PSFT_OLTP_VER=91      |

The default value for this parameter is PSFT\_OLTP\_VER=91.



---



---

**Caution:** You must enter the correct value for this parameter or else the upgrade process will fail.

---



---

## A.2.4 Setting Parameters in 7963\_UPG\_PARAMS.txt for Siebel Industry Applications Source Systems

This procedure is applicable when you are upgrading to Oracle BI Applications release 7.9.6.3 and the source system is Siebel Industry Applications.

### To set parameters in the 7963\_UPG\_PARAMS.txt file for Siebel Industry Applications source systems

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the file 7963\_UPG\_PARAMS.txt.
3. Search for the parameter \$\$VERTICAL\_UPGRADE.
4. If you are using Siebel Industry Applications, set the value to 1.  
For example: \$\$VERTICAL\_UPGRADE=1  
If you are not using Siebel Industry Applications, leave the default value (0).
5. Save the 7963\_UPG\_PARAMS.txt file.

## A.3 Setting Application-Specific Parameters in Informatica Parameter Files

This section provides information about setting parameters in the Informatica parameter files that are specific to an Oracle BI Applications application. You may need to set or update these parameters depending on your environment. The topic headings below indicate the application and version of the parameter file that require configuration.

This appendix includes the following topics:

- [Section A.3.1, "Setting Parameters and Mappings in 796\\_UPG\\_PARAMS.txt for Oracle Financial Analytics"](#)
- [Section A.3.2, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Project Analytics"](#)
- [Section A.3.3, "Setting Parameters in 796\\_UPG\\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics"](#)
- [Section A.3.4, "Setting Parameters and Mappings in 7962\\_UPG\\_PARAMS.txt for Oracle Financial Analytics"](#)
- [Section A.3.5, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Financial Analytics"](#)
- [Section A.3.6, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Human Resources Analytics"](#)
- [Section A.3.7, "Setting Parameters in 7963\\_UPG\\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System"](#)

## A.3.1 Setting Parameters and Mappings in 796\_UPG\_PARAMS.txt for Oracle Financial Analytics

This section provides information about parameters and mappings specific to Oracle Financial Analytics that you may need to configure or update depending on your environment. The procedures in this section are applicable for upgrades to Oracle BI Applications release 7.9.6.

This section includes the following topics:

- [Section A.3.1.1, "Setting Parameters for Value Set Hierarchies and FSG Hierarchies"](#)
- [Section A.3.1.2, "Setting Parameters for GL Data Extraction"](#)
- [Section A.3.1.3, "Setting the COGS Fact Mapping for Oracle EBS R12"](#)
- [Section A.3.1.4, "Setting the \\$\\$Hint1 Parameter for Oracle Databases"](#)

### A.3.1.1 Setting Parameters for Value Set Hierarchies and FSG Hierarchies

In Oracle Financial Analytics, the default behavior is for Value Set Hierarchies to be enabled and Financial Statement Generator (FSG) Hierarchies to be disabled.

If you have changed this behavior by disabling Value Set Hierarchies and enabling FSG Hierarchies, then you need to set the parameters that control this behavior in the 796\_UPG\_PARAMS.txt file and the DAC configuration tags.

#### To set the FSG Hierarchies and Value Set Hierarchies parameters

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 796\_UPG\_PARAMS.txt file.
3. Search for the parameter \$\$IS\_FSGHIERARCHY\_INSTALLED, and set the value to Y.

For example:

```
$$IS_FSGHIERARCHY_INSTALLED=Y
```

4. Search for the parameter \$\$IS\_VALUESETHIERARCHY\_INSTALLED, and set the value to N.

For example:

```
$$IS_VALUESETHIERARCHY_INSTALLED=N
```

### A.3.1.2 Setting Parameters for GL Data Extraction

If you have configured your GL data extraction to limit the set of books or ledgers extracted, you need to set the appropriate parameters in the 796\_UPG\_PARAMS.txt file to match the same parameters in the DAC source system parameters.

This section includes the following topics:

- [Section A.3.1.2.1, "Setting GL Data Extraction Parameters for Oracle EBS 11i Sources"](#)
- [Section A.3.1.2.2, "Setting GL Data Extraction Parameters for Oracle EBS R12 Sources"](#)

**A.3.1.2.1 Setting GL Data Extraction Parameters for Oracle EBS 11i Sources** Follow this procedure to set GL data extraction parameters for Oracle EBS 11i sources.

#### To set GL data extraction parameters

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 796\_UPG\_PARAMS.txt file.
3. Note the values for the following parameters:
  - \$\$FILTER\_BY\_SET\_OF\_BOOKS\_ID
  - \$\$FILTER\_BY\_SET\_OF\_BOOKS\_TYPE
  - \$\$SET\_OF\_BOOKS\_ID\_LIST
  - \$\$SET\_OF\_BOOKS\_TYPE\_LIST
4. In DAC, go to the Design view, and select the appropriate custom container.
5. Select the Source System Parameters tab.
6. Query for the parameters listed in Step 3, and compare the values.
7. If necessary, change the values for the parameters in the 796\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
8. Save the 796\_UPG\_PARAMS.txt file.

**A.3.1.2.2 Setting GL Data Extraction Parameters for Oracle EBS R12 Sources** Follow this procedure to configure GL data extraction parameters for Oracle EBS R12 sources.

**To set GL data extraction parameters**

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 796\_UPG\_PARAMS.txt file.
3. Note the values for the following parameters:
  - \$\$FILTER\_BY\_LEDGER\_ID
  - \$\$FILTER\_BY\_LEDGER\_TYPE
  - \$\$LEDGER\_ID\_LIST
  - \$\$LEDGER\_TYPE\_LIST
4. In DAC, go to the Design view, and select the appropriate custom container.
5. Select the Source System Parameters tab.
6. Query for the parameters listed in Step 3, and compare the values.
7. If necessary, change the values for the parameters in the 796\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
8. Save the 796\_UPG\_PARAMS.txt file.

**A.3.1.3 Setting the COGS Fact Mapping for Oracle EBS R12**

For Oracle EBS R12 sources, follow this procedure to update the COGS fact mapping.

**To set the COGS fact mapping**

1. Launch Informatica PowerCenter Designer.
2. Navigate to the folder UPGRADE\_7951\_TO\_796\_ORA12.
3. Open the mapping SDE\_ORA\_GLCOGSFact\_UPG796.

4. Open the maplet `mplt_BC_ORA_GLCOGSFact`.
5. Open the Source Qualifier Transformation, do the following:
  - a. Open the SQL Query property.
  - b. In the WHERE clause of the query locate the hard-coded filter on the Transaction Type ID and Transaction Action ID.  
  
For example:

```
MMT.TRANSACTION_TYPE_ID IN (15, 33, 10008) AND
(MMT.TRANSACTION_ACTION_ID, MTA.ACCOUNTING_LINE_TYPE) IN
((27, 2), (1, 36), (36, 35))
```
6. Change the values to the actual values you used in the `SDE_ORA_GLCOGSFact` mapping in the base Informatica Repository code.

#### A.3.1.4 Setting the \$\$Hint1 Parameter for Oracle Databases

If your target data warehouse is an Oracle database server, follow this procedure to set the `$$Hint1` parameter.

##### To set the \$\$Hint1 parameter

1. Navigate to the `SrcFiles` folder on the Informatica Server machine, for example, `<INFA_HOME>\server\infa_shared\SrcFiles`.
2. Open the `796_UPG_PARAMS.txt` file.
3. Locate the `$$Hint1` parameter for the appropriate version of the Oracle database, and enter the following value:

```
/*+ USE_HASH(W_GL_BALANCE_F, W_GL_ACCOUNT_D, W_GLACCT_SEG_
CONFIG_TMP) */
```

For example:

```
mplt_GLBalanceAggrByAcctSegCodes.$$Hint1=/*+ USE_HASH(W_GL_
BALANCE_F, W_GL_ACCOUNT_D, W_GLACCT_SEG_CONFIG_TMP) */
```

4. Save the `796_UPG_PARAMS.txt` file.

#### A.3.2 Setting Parameters in 796\_UPG\_PARAMS.txt for Oracle Project Analytics

If you are deploying Oracle Project Analytics and you are upgrading to Oracle BI Applications release 7.9.6, follow the procedure in this section to configure the `IS_PROJECTS_INSTALLED` parameter.

##### To set the IS\_PROJECTS\_INSTALLED parameter

1. Navigate to the `SrcFiles` folder on the Informatica Server machine, for example, `<INFA_HOME>\server\infa_shared\SrcFiles`.
2. Open the `796_UPG_PARAMS.txt` file.
3. Search for the parameter `$$IS_PROJECTS_INSTALLED`, and set the value to `Y`.

For example:

```
$$IS_PROJECTS_INSTALLED=Y
```

4. Save the `796_UPG_PARAMS.txt` file.

### A.3.3 Setting Parameters in 796\_UPG\_PARAMS.txt for Oracle Supply Chain and Order Management Analytics

If you are deploying Oracle Supply Chain and Order Management Analytics and you are upgrading to Oracle BI Applications release 7.9.6, follow the procedures in this section to set parameters in 796\_UPG\_PARAM.txt.

This section includes the following topics:

- [Section A.3.3.1, "Setting the TIME\\_GRAIN Parameter for Sales Order Lines Aggregate Fact and Invoice Lines Aggregate Fact Tables"](#)
- [Section A.3.3.2, "Setting the \\$\\$PERIOD Parameter for the Customer Status History Fact Table"](#)

#### A.3.3.1 Setting the TIME\_GRAIN Parameter for Sales Order Lines Aggregate Fact and Invoice Lines Aggregate Fact Tables

If you are deploying Oracle Supply Chain and Order Management Analytics, you need to configure the TIME\_GRAIN parameter for the Sales Order Lines Aggregate Fact table and for the Invoice Lines Aggregate Fact table. For instructions, see the section titled, "Process of Aggregating Oracle Supply Chain and Order Management," in *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*.

#### A.3.3.2 Setting the \$\$PERIOD Parameter for the Customer Status History Fact Table

If you are deploying Oracle Supply Chain and Order Management Analytics, you need to configure the \$\$PERIOD parameter for the Customer Status History Fact table. For instructions, see the section titled, "How to Configure the Customer Status History Fact Table," in *Oracle Business Intelligence Applications Configuration Guide for Informatica PowerCenter Users*.

### A.3.4 Setting Parameters and Mappings in 7962\_UPG\_PARAMS.txt for Oracle Financial Analytics

If you are deploying Oracle Financial Analytics and you are upgrading to Oracle BI Applications release 7.9.6.2, follow the procedures in this section for your particular source system to set parameters in 7962\_UPG\_PARAM.txt.

This section includes the following topics:

- [Section A.3.4.1, "Setting GL Data Extraction Parameters for Oracle EBS 11i Sources"](#)
- [Section A.3.4.2, "Setting GL Data Extraction Parameters for Oracle EBS 12 Sources"](#)
- [Section A.3.4.3, "Setting GL Data Extraction Parameters for PeopleSoft Sources"](#)
- [Section A.3.4.4, "Setting the COGS Fact Mapping for Oracle EBS R12"](#)

#### A.3.4.1 Setting GL Data Extraction Parameters for Oracle EBS 11i Sources

Follow this procedure to configure GL data extraction parameters for Oracle EBS 11i sources.

##### To set GL data extraction parameters

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7962\_UPG\_PARAMS.txt file.
3. Note the values for the following parameters:

- mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$FILTER\_BY\_SET\_OF\_BOOKS\_ID
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$FILTER\_BY\_SET\_OF\_BOOKS\_TYPE
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$SET\_OF\_BOOKS\_ID\_LIST
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$SET\_OF\_BOOKS\_TYPE\_LIST
4. In DAC, go to the Design view, and select the appropriate custom container.
  5. Select the Source System Parameters tab.
  6. Query for the parameters listed in Step 3, and compare the values.
  7. If necessary, change the values for the parameters in the 7962\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
  8. Save the 7962\_UPG\_PARAMS.txt file.

#### **A.3.4.2 Setting GL Data Extraction Parameters for Oracle EBS 12 Sources**

Follow this procedure to configure GL data extraction parameters for Oracle EBS 12 sources.

##### **To set GL data extraction parameters**

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7962\_UPG\_PARAMS.txt file.
3. Note the values for the following parameters:
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$FILTER\_BY\_LEDGER\_ID
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$FILTER\_BY\_LEDGER\_TYPE
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$LEDGER\_ID\_LIST
  - mplt\_BC\_ORA\_GL\_Balance\_Fact.\$\$LEDGER\_TYPE\_LIST
4. In DAC, go to the Design view, and select the appropriate custom container.
5. Select the Source System Parameters tab.
6. Query for the parameters listed in Step 3, and compare the values.
7. If necessary, change the values for the parameters in the 7962\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.
8. Save the 7962\_UPG\_PARAMS.txt file.

#### **A.3.4.3 Setting GL Data Extraction Parameters for PeopleSoft Sources**

Follow this procedure to configure GL data extraction parameters for PeopleSoft 8.8, 8.9, and 9.0 sources.

##### **To set GL data extraction parameters**

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7962\_UPG\_PARAMS.txt file.
3. Note the values for the following parameters:
  - mplt\_BC\_PSFT\_GLBalaceFact.\$\$FISCAL\_YEAR

- \$\$LOW\_DATE
4. In DAC, go to the Design view, and select the appropriate custom container.
  5. Select the Source System Parameters tab.
  6. Query for the parameters listed in Step 3, and compare the values.
  7. If necessary, change the values for the parameters in the 7962\_UPG\_PARAMS.txt file to match the values for the parameters in the DAC Source System Parameters tab.

---

**Note:** The parameter \$\$FISCAL\_YEAR is fiscal year from which you want to extract the GL Balances into the data warehouse. GL Balances from PS\_LEDGER will be extracted from this year forward into the data warehouse. All balances prior to this year will not be extracted.

---

8. Save the 7962\_UPG\_PARAMS.txt file.

#### A.3.4.4 Setting the COGS Fact Mapping for Oracle EBS R12

For Oracle EBS R12 sources, follow this procedure to update the COGS fact mapping.

##### To set the COGS fact mapping

1. Launch Informatica PowerCenter Designer.
2. Navigate to the folder UPGRADE\_7961\_to\_7962\_ORAR12.
3. Open the mapping SDE\_ORA\_GLCOGSFact\_UPG7962.
4. Open the mapplet mplt\_BC\_ORA\_GLCOGSFact.
5. Open the Source Qualifier Transformation, and do the following:
  - a. Open the SQL Query property.
  - b. In the WHERE clause of the query, locate the hard-coded filter on the Transaction Type ID and Transaction Action ID.

For example:

```
MMT.TRANSACTION_TYPE_ID IN (15, 33, 10008) AND
(MMT.TRANSACTION_ACTION_ID, MTA.ACCOUNTING_LINE_TYPE) IN
((27, 2), (1, 36), (36, 35))
```

6. Change the values to the actual values you used in the SDE\_ORA\_GLCOGSFact mapping in the main Informatica Repository code.

### A.3.5 Setting Parameters in 7963\_UPG\_PARAMS.txt for Oracle Financial Analytics

If you are deploying Oracle Financial Analytics and you are upgrading to Oracle BI Applications release 7.9.6.3, follow the procedures in this section for your particular source system to set parameters in 7963\_UPG\_PARAM.txt.

This section includes the following topics:

- [Section A.3.5.1, "Setting Parameters for All Source Systems"](#)
- [Section A.3.5.2, "Setting Parameters Specific to PeopleSoft"](#)
- [Section A.3.5.3, "Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J"](#)

### A.3.5.1 Setting Parameters for All Source Systems

This section provides instructions for setting parameters in the 7963\_UPG\_PARAMS.txt file for Oracle Financial Analytics for all source systems, including Oracle EBS 11i, Oracle EBS 12, PeopleSoft, and JD Edwards.

#### To set the soft delete parameter

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7963\_UPG\_PARAMS.txt file.
3. Note the value for the parameter `$$IS_SOFT_DELETE_IMPLEMENTED`.

The default value for this parameter is `$$IS_SOFT_DELETE_IMPLEMENTED=N`.

If the Soft Delete parameter is implemented for any of the Oracle Financial Analytics modules, for example, GL, AP, AR, and Profitability (COGS/Revenue), you must change the parameter value to `$$IS_SOFT_DELETE_IMPLEMENTED=Y` for the appropriate module for which it is implemented.

4. Save the 7963\_UPG\_PARAMS.txt file.

### A.3.5.2 Setting Parameters Specific to PeopleSoft

This section provides instructions for setting a parameter for Oracle Financial Analytics in the 7963\_UPG\_PARAMS.txt file that is specific to PeopleSoft source systems.

#### To set the parameter specific to PeopleSoft source systems

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7963\_UPG\_PARAMS.txt file.
3. The parameter `$$PSFT_OLTP_VER` specifies the version of the PeopleSoft OLTP. The default value for this parameter is `$$PSFT_OLTP_VER=90`.

Change the parameter value to match the appropriate version of the PeopleSoft OLTP. For example:

- For PeopleSoft version 8.9, enter `$$PSFT_OLTP_VER=89`.
- For PeopleSoft version 9.0, enter `$$PSFT_OLTP_VER=90`.
- For PeopleSoft version 9.1, enter `$$PSFT_OLTP_VER=91`.

**Note:** This parameter is needed to run the Transaction type dimension maps, which are missing in the PeopleSoft 8.9 source system. Therefore, setting this parameter is particularly important if you are using PeopleSoft 8.9.

4. Save the 7963\_UPG\_PARAMS.txt file.

### A.3.5.3 Setting Parameters Specific to Oracle EBS 11.5.10 Family Pack OIE.I and OIE.J

This section provides instructions for setting a parameter for Oracle Financial Analytics in the 7963\_UPG\_PARAMS.txt file that is specific to Oracle EBS 11.5.10 family pack OIE.I and OIE.J.

#### To set the parameter specific to Oracle EBS 11.5.10 OIE.I or OIE.J

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.



2. Open the 7963\_UPG\_PARAMS.txt file.
3. The parameter \$\$ORA\_OLTP\_VER specifies the version of the Oracle EBS OLTP. Change the parameter value to match the appropriate version of the family pack for your Oracle EBS 11.5.10 environment.
  - For Oracle EBS 11.5.10 family pack OIE.I, enter \$\$ORA\_OLTP\_VER=OIEI.
  - For Oracle EBS 11.5.10 family pack OIE.J, enter \$\$ORA\_OLTP\_VER=OIEJ.
4. Save the 7963\_UPG\_PARAMS.txt file.

### A.3.6 Setting Parameters in 7963\_UPG\_PARAMS.txt for Oracle Human Resources Analytics

If you are deploying Oracle Human Resources Analytics and you are upgrading to Oracle BI Applications release 7.9.6.3, follow the procedure in this section to set parameters in 7963\_UPG\_PARAM.txt.

#### To set the parameter for Oracle Human Resources

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7963\_UPG\_PARAMS.txt file.
3. Search for the parameter \$\$HR\_WRKFC\_EXTRACT\_DATE.  
The parameter value is defined with the "TO\_DATE" function.
4. Edit the function to use the appropriate function for data conversion based on the database type:
  - For Oracle databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For SQL Server databases, use "CONVERT(datetime, '1899-01-01')"
  - For DB2 databases, use "TO\_DATE('1899-01-01 00:00:00', 'YYYY-MM-DD HH24:MI:SS')"
  - For Teradata databases, use "cast('1899-01-01' as timestamp format 'YYYY-MM-DD')"
5. Save the 7963\_UPG\_PARAMS.txt file.

### A.3.7 Setting Parameters in 7963\_UPG\_PARAMS.txt for Oracle Procurement and Spend Analytics on PeopleSoft Source System

Follow the procedure in this section if you are deploying Oracle Procurement and Spend Analytics on a PeopleSoft 9.0 source system.

#### To set the parameters for Oracle Procurement and Spend Analytics

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7963\_UPG\_PARAMS.txt file.
3. Set the parameter \$\$PSFT\_DATASOURCE\_NUM\_ID to the relevant value from the source system setup.
4. Save the 7963\_UPG\_PARAMS.txt file.

### A.3.8 Setting Parameters in 7963\_UPG\_PARAMS.txt for Oracle Procurement and Spend Analytics on Oracle EBS 11i and 12 Source Systems

Follow the procedure in this section if you are deploying Oracle Procurement and Spend Analytics on a Oracle EBS 11i and 12 source systems.

#### To set the parameters for Oracle Procurement and Spend Analytics

1. Navigate to the SrcFiles folder on the Informatica Server machine, for example, <INFA\_HOME>\server\infa\_shared\SrcFiles.
2. Open the 7963\_UPG\_PARAMS.txt file.
3. Search for the entry:

```
[UPGRADE_7962_to_7963_ORA<version>.SIL_PurchaseCycleLinesFact_UPG7963]
```

4. Set the parameter **\$\$ORA\_DATASOURCE\_NUM\_ID** to the relevant value from the source system setup.

The default value for Oracle EBS 11i is

```
$$ORA_DATASOURCE_NUM_ID_LIST= (4)
```

The default value for Oracle EBS 12is

```
$$ORA_DATASOURCE_NUM_ID_LIST= (9)
```

5. Save the 7963\_UPG\_PARAMS.txt file.

---

---

## Generating and Running Scripts to Drop and Create Indexes

This appendix provides instructions for generating and running scripts to drop and create indexes while upgrading to releases 7.9.6.2 and 7.9.6.3.

This appendix contains the following topics:

- [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2"](#)
- [Section B.2, "Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2"](#)
- [Section B.3, "Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2"](#)
- [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3"](#)
- [Section B.5, "Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3"](#)

### B.1 Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2

This section provides instructions for generating a script to drop indexes as well as scripts to create ETL indexes and query indexes. This procedure is applicable when you are upgrading to release 7.9.6.2.

#### To generate a script to drop indexes and a script to create indexes

1. Go to the folder OracleBI\dwrep\Upgrade\CTLFiles.
2. Open the UPGRADE\_7961\_to\_7962\_<source system>.txt file.  
This file contains a list of data warehouse tables that were upgraded for this release.
3. Open a text editor, and copy and paste the contents of UPGRADE\_7961\_to\_7962\_<source system>.txt into a new file.

Note that table names are separated by a comma (,).

4. Find and replace the comma (,) with OR.

Make sure there is only one space before and after OR. The format of the text string should appear as follows:

```
table_1 OR table_2 OR table_3 OR table_4
```

For example:

```
W_POSITION_D OR W_PURCH_COST_F OR W_EXPENSE_F OR W_PURCH_
CYCLE_LINE_A OR W_ORA_GL_LINKAGE_INFO_REVN_TMP OR W_
POSITION_DH
```

5. Open the DAC repository for the release from which you are upgrading. For example, if you are upgrading from release 7.9.6.1 to 7.9.6.2, you would open the release 7.9.6.1 DAC repository
6. From the DAC toolbar, select the appropriate source system container from the drop-down list.
7. In the Design view, click the Indices tab.
8. In the top pane toolbar, click Query.  
The top pane window now displays Query mode.
9. Copy the text string from the text file you created in step 5 into the Table Name field.
10. Click Go.
11. In the list of query results, right-click, and then select Generate Index Scripts.
12. Select "All records in the list."
13. In the Generate Index Scripts dialog box, do one of the following:

- To generate index scripts based on a physical data source connection that you previously set up, select the "Use source information" check box, and then select the appropriate physical data source connection from the drop-down list.
- To generate index scripts based on the data warehouse database type, deselect the "Use source information" check box, and then select the appropriate database type.

If the database type is Oracle, enter a table owner, and optionally enter a tablespace in which the indexes will be dropped and created.

If the database type is SQL Server, DB2 or Teradata, enter a table owner.

14. Click OK.

A message box states whether the scripts were successfully generated. The following scripts are saved in the folder \bifoundation\dac\log\scripts:

- dropAllIndexes.sql
- createETLIndexes.sql
- createQueryIndexes.sql

## B.2 Running the Script to Drop All Indexes for Upgrading to Release 7.9.6.2

This section provides instructions for running the dropAllIndexes.sql script to drop all indexes during the upgrade to release 7.9.6.2. Before you perform this procedure, you must first generate the script, as described in [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)

**To run the script to drop all indexes**

1. Open the SQL client for your database type.
2. Navigate to the folder \bifoundation\dac\log\scripts.
3. Open the dropAllIndexes.sql file, and copy the contents into the SQL client.
4. Execute the script.

## B.3 Running the Scripts to Create ETL and Query Indexes for Upgrading to Release 7.9.6.2

This section provides instructions for running the createETLIndexes.sql and createQueryIndexes.sql scripts to create indexes during the upgrade to release 7.9.6.2. Before you perform this procedure, you must first generate the scripts, as described in [Section B.1, "Generating Scripts to Drop and Create Indexes for Upgrading to Release 7.9.6.2."](#)

### To run the script to create ETL and Query indexes

1. Open the SQL client for your database type.
2. Navigate to the folder \bifoundation\dac\log\scripts.
3. Open either the createETLIndexes.sql script or the createQueryIndexes.sql script, depending on the upgrade step you are on.

Note: You must run the createETLIndexes.sql script before you run the Informatica workflows and the createQueryindexes.script after you run the Informatica workflows.

4. Copy the contents of the SQL script into the SQL client.
5. Execute the script.

## B.4 Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3

This section provides instructions for generating scripts to drop and create query indexes. This procedure is applicable when you are upgrading to release 7.9.6.3.

### To generate a scripts to drop and created query indexes

1. Go to the folder OracleBI\dwrep\Upgrade\CTLFiles.
2. Open the UPGRADE\_7962\_to\_7963\_<source system>.txt file.

This file contains a list of data warehouse tables that were upgraded for this release.

3. Open a text editor, and copy and paste the contents of UPGRADE\_7962\_to\_7963\_<source system>.txt into a new file.

Note that table names are separated by a comma (,).

4. Find and replace the comma (,) with OR.

Make sure there is only one space before and after OR. The format of the text string should appear as follows:

```
table_1 OR table_2 OR table_3 OR table_4
```

For example:

W\_POSITION\_D OR W\_PURCH\_COST\_F OR W\_EXPENSE\_F OR W\_PURCH\_CYCLE\_LINE\_A OR W\_ORA\_GL\_LINKAGE\_INFO\_REVN\_TMP OR W\_POSITION\_DH

5. Open the DAC repository for the release from which you are upgrading. For example, if you are upgrading from release 7.9.6.2 to 7.9.6.3, you would open the release 7.9.6.2 DAC repository
6. From the DAC toolbar, select the appropriate source system container from the drop-down list.
7. In the Design view, click the Indices tab.
8. In the top pane toolbar, click Query.  
The top pane window now displays Query mode.
9. Copy the text string from the text file you created in step 5 into the Table Name field.
10. Click Go.
11. In the list of query results, right-click, and then select Generate Index Scripts.
12. Select "All records in the list."
13. In the Generate Index Scripts dialog box, do one of the following:
  - To generate index scripts based on a physical data source connection that you previously set up, select the "Use source information" check box, and then select the appropriate physical data source connection from the drop-down list.
  - To generate index scripts based on the data warehouse database type, deselect the "Use source information" check box, and then select the appropriate database type.  
  
If the database type is Oracle, enter a table owner, and optionally enter a tablespace in which the indexes will be dropped and created.  
  
If the database type is SQL Server, DB2 or Teradata, enter a table owner.
14. Click OK.  
  
A message box states whether the scripts were successfully generated. The following scripts are saved in the folder \bifoundation\dac\log\scripts:
  - dropQueryIndices.sql
  - createQueryIndices.sql

**Note:** Additional files may be generated; however, you will not need to run them.

## B.5 Running the Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3

This section provides instructions for running the dropQueryIndices.sql and createQueryIndices.sql scripts when upgrading to release 7.9.6.3. Before you perform this procedure, you must first generate the scripts, as described in [Section B.4, "Generating Scripts to Drop and Create Query Indexes for Upgrading to Release 7.9.6.3."](#)

**To run the script to drop or create query indexes**

1. Open the SQL client for your database type.
2. Navigate to the folder `\bifoundation\dac\log\scripts`.
3. Open either the `dropQueryIndices.sql` script or the `createQueryIndices.sql` script, depending on the upgrade step you are on.
4. Copy the contents of the SQL script into the SQL client.
5. Execute the script.





---

# Creating Indexes for Oracle Financial Analytics

This appendix provides information about indexes that should be created to improve performance when using Oracle Financial Analytics. The indexes discussed in this appendix are applicable to Oracle BI Applications release 7.9.6.2.

This appendix includes the following topics:

- [Section C.1, "Indexes for SIL\\_PositionDimension\\_UPD7962 Mapping"](#)
- [Section C.2, "Indexes for SIL\\_PositionDimensionHierarchy\\_UPD7962 Mapping"](#)

## C.1 Indexes for SIL\_PositionDimension\_UPD7962 Mapping

Make sure the following indexes are created for the SIL\_PositionDimension\_UPD7962 mapping:

- Primary key index on the ROW\_WID column on table W\_POSITION\_D
- Normal index on the INTEGRATION\_ID and DATASOURCE\_NUM\_ID columns on table W\_POSITION\_D\_7962

## C.2 Indexes for SIL\_PositionDimensionHierarchy\_UPD7962 Mapping

Make sure the following indexes are created for the SIL\_PositionDimensionHierarchy\_UPD7962 mapping:

- Primary key index on ROW\_WID column on table W\_POSITION\_D
- Primary key index on ROW\_WID column on table W\_POSITION\_DH
- Primary key index on ROW\_WID column on table W\_POSITION\_DH\_7962
- Unique index on INTEGRATION\_ID, DATASOURCE\_NUM\_ID, and EFFECTIVE\_FROM\_DT columns on table W\_POSITION\_D
- Unique index on BASE\_POSTN\_ID, DATASOURCE\_NUM\_ID, and EFFECTIVE\_FROM\_DT columns on table W\_POSITION\_DH\_7962



# D

---

---

## Setting Database-Specific Parameters for the reset\_infa\_seq\_gen.bat File

This appendix contains the following topics:

- Section D.1, "Running reset\_infa\_seq\_gen.bat on Base Informatica Repository"
- Section D.2, "Running reset\_infa\_seq\_gen\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle"
- Section D.3, "Running reset\_infa\_seq\_gen\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server"
- Section D.4, "Running reset\_infa\_seq\_gen.bat on Upgrade Informatica Repository"
- Section D.5, "Running reset\_infa\_seq\_gen\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle"
- Section D.6, "Running reset\_infa\_seq\_gen\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server"

### D.1 Running reset\_infa\_seq\_gen.bat on Base Informatica Repository

Follow this procedure to run the reset\_infa\_seq\_gen.bat file on the Base Informatica Repository if your Oracle Business Analytics Warehouse runs on an Oracle, DB2, or MSSQL Server database.

This section contains the following topics:

- Section D.1.1, "Entering Values for the reset\_infa\_seq\_gen.bat File on the Base Informatica Repository"
- Section D.1.2, "Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D"
- Section D.1.3, "Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D"

#### D.1.1 Entering Values for the reset\_infa\_seq\_gen.bat File on the Base Informatica Repository

Follow this procedure to enter values for the reset\_infa\_seq\_gen.bat file to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\- 2. Open the reset\_infa\_seq\_gen.bat file.

3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific). See one of the following tables:
  - [Table D-1, " Oracle Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)
  - [Table D-2, " SQL Server Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)
  - [Table D-3, " IBM DB2 Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.1.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.1.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen.bat until you have completed the procedures in either [Section D.1.2](#) or [Section D.1.3](#).

### Oracle Database Parameter Settings for the Base Informatica Repository

**Table D-1 Oracle Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                                | Setting                                                                                                                                                                                                                                                               |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP USERNAME                            | Enter the data warehouse database user ID.                                                                                                                                                                                                                            |
| OLAP PASSWORD                            | Enter the user ID/password@connection string of the data warehouse database.                                                                                                                                                                                          |
| OLAP TNS ENTRY                           | Enter the TNS entry for the data warehouse database.                                                                                                                                                                                                                  |
| INFORMATICA SERVER ADDRESS               | Enter the host address for the Informatica Service.                                                                                                                                                                                                                   |
| INFORMATICA REPOSITORY DATABASE TNSENTRY | Enter the TNS entry for the Informatica Repository database.                                                                                                                                                                                                          |
| INFORMATICA REPOSITORY DB USERNAME       | Enter the user ID of the base Informatica Repository database.                                                                                                                                                                                                        |
| INFORMATICA REPOSITORY DB PASSWORD       | Enter the password@connection string of the base Informatica Repository database.                                                                                                                                                                                     |
| INFORMATICA REPOSITORY                   | Enter the name of the base Informatica Repository.                                                                                                                                                                                                                    |
| INFORMATICA REPOSITORY USERNAME          | Enter the user ID of the base Informatica Repository.                                                                                                                                                                                                                 |
| INFORMATICA REPOSITORY PASSWORD          | Enter the password for the base Informatica Repository.                                                                                                                                                                                                               |
| INFORMATICA FODLER                       | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |

**Table D–1 (Cont.) Oracle Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter        | Setting                                                                |
|------------------|------------------------------------------------------------------------|
| INFORMATICA_PORT | Enter the port number for the Repository Service. The default is 6001. |

### SQL Server Database Parameter Settings for the Base Informatica Repository

**Table D–2 SQL Server Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                                | Setting                                                                                                                                                                                                                                                           |
|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP_SERVER                              | Enter the name of the SQL Server for the OLAP database.                                                                                                                                                                                                           |
| OLAP_DATABASE                            | Enter the name of the database on which OLAP data is available.                                                                                                                                                                                                   |
| OLAP_USERNAME                            | Enter the user ID of the data warehouse database.                                                                                                                                                                                                                 |
| OLAP_PASSWORD                            | Enter the user ID/password@connection string of the data warehouse database.                                                                                                                                                                                      |
| INFORMATICA_REPOSITORY_DB_SERVER_ADDRESS | Enter the Informatica Server host address.                                                                                                                                                                                                                        |
| INFORMATICA_DATABASE                     | Enter the name of the database in which the base Informatica Repository metadata is available.                                                                                                                                                                    |
| INFORMATICA_DB_USERNAME                  | Enter the user ID of the base Informatica Repository database.                                                                                                                                                                                                    |
| INFORMATICA_DB_PASSWORD                  | Enter the password @connection string of the base Informatica Repository database.                                                                                                                                                                                |
| INFORMATICA_REPOSITORY_SERVER_ADDRESS    | Enter the name of the machine that runs the Repository Service for the base Repository.                                                                                                                                                                           |
| INFORMATICA_REPOSITORY_USERNAME          | Enter the user ID of the base Informatica Repository.                                                                                                                                                                                                             |
| INFORMATICA_REPOSITORY_PASSWORD          | Enter the password of the base Informatica Repository.                                                                                                                                                                                                            |
| INFORMATICA_FOLDER                       | For all source systems except Siebel Industry Applications, enter the value HOR.<br>For Siebel Industry Applications, you need to run reset_infa_seq_gen.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                         | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                            |

### IBM DB2 Database Parameter Settings for the Base Informatica Repository

**Table D–3 IBM DB2 Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter     | Setting                                                                      |
|---------------|------------------------------------------------------------------------------|
| OLAP_USERNAME | Enter the user ID of the data warehouse database.                            |
| OLAP_PASSWORD | Enter the user ID/password@connection string of the data warehouse database. |

**Table D-3 (Cont.) IBM DB2 Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                             | Setting                                                                                                                                                                                                                                                               |
|---------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| INFA_DATABASE<br>ALIAS                | Enter the name of the database in which Informatica Repository metadata is available.                                                                                                                                                                                 |
| INFORMATICA_DB<br>USERNAME            | Enter the user ID of the base Informatica Repository database.                                                                                                                                                                                                        |
| INFORMATICA_DB<br>PASSWORD            | Enter the password @connection string of the base Informatica Repository database.                                                                                                                                                                                    |
| INFORMATICA<br>REPOSITORY             | Enter the name of the base Informatica Repository.                                                                                                                                                                                                                    |
| INFORMATICA<br>REPOSITORY<br>USERNAME | Enter the user ID of the base Informatica Repository.                                                                                                                                                                                                                 |
| INFORMATICA<br>REPOSITORY<br>PASSWORD | Enter the password for the base Informatica Repository.                                                                                                                                                                                                               |
| INFORMATICA_SERVER<br>ADDRESS         | Enter the Informatica Server host address.                                                                                                                                                                                                                            |
| INFORMATICA_FOLDER                    | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                      | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                                |

## D.1.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.1.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```
  - d. In the reset\_infa\_seq\_gen.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.

3. Execute reset\_infa\_seq\_gen.bat.

### D.1.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.1.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:

- a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. In the code example, note the value 1234567. You will replace this value in a following step.

- c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```

- d. In the reset\_infa\_seq\_gen.bat file, replace the value 1234567 with the value that was returned by the SQL statement.

2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_
SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1
>>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2
>>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.

- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.

3. Execute reset\_infa\_seq\_gen.bat.

## D.2 Running reset\_infa\_seq\_gen\_ORA.bat on Base Repository for DW on Teradata and Repository on Oracle

Follow this procedure if your Oracle Business Analytics Warehouse is on a Teradata database and your Informatica Repository is on an Oracle database.

This section contains the following topics:

- [Section D.2.1, "Entering Values for the reset\\_infa\\_seq\\_gen\\_ORA.bat File on the Base Informatica Repository"](#)
- [Section D.2.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)
- [Section D.2.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)

### D.2.1 Entering Values for the reset\_infa\_seq\_gen\_ORA.bat File on the Base Informatica Repository

Follow this procedure to enter values for the reset\_infa\_seq\_gen\_ORA.bat file to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).



1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\Teradata.
2. Open the reset\_infa\_seq\_gen\_ORA.bat file.
3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific).

**Table D-4 Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_ORA.bat**

| Parameter                          | Setting                                                                                                                                                                                                                                                                   |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP_SERVER                        | Enter the name of the Teradata Server for the data warehouse database.                                                                                                                                                                                                    |
| OLAP_DATABASE                      | Enter the name of the database on which OLAP data is available.                                                                                                                                                                                                           |
| OLAP_USERNAME                      | Enter the user ID of the data warehouse database.                                                                                                                                                                                                                         |
| OLAP_PASSWORD                      | Enter the password of the data warehouse database.                                                                                                                                                                                                                        |
| INFORMATICA_SERVER_ADDRESS         | Enter the Informatica Server host address.                                                                                                                                                                                                                                |
| INFORMATICA_REPOSITORY_DB_USERNAME | Enter the user ID of the base Informatica Repository database.                                                                                                                                                                                                            |
| INFORMATICA_REPOSITORY_DB_PASSWORD | Enter the password @connection string of the base Informatica Repository database.                                                                                                                                                                                        |
| INFORMATICA_REPOSITORY             | Enter the name of the base Informatica Repository.                                                                                                                                                                                                                        |
| INFORMATICA_REPOSITORY_USERNAME    | Enter the user ID of the base Informatica Repository.                                                                                                                                                                                                                     |
| INFORMATICA_REPOSITORY_PASSWORD    | Enter the password for the base Informatica Repository.                                                                                                                                                                                                                   |
| INFORMATICA_FOLDER                 | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen_ORA.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                   | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                                    |

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.2.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.2.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen\_ORA.bat until you have completed the procedures in either [Section D.2.2](#) or [Section D.2.3](#).

## D.2.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.2.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```

- d. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
  - d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.

3. Execute reset\_infa\_seq\_gen\_ORA.bat.

### D.2.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.2.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:

- a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. In the code example, note the value 1234567. You will replace this value in a following step.
- c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```

- d. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the value 1234567 with the value that was returned by the SQL statement.

2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen\_ORA.bat.

## D.3 Running reset\_infa\_seq\_gen\_MSSQL.bat on Base Repository for DW on Teradata and Repository on MSSQL Server

Follow this procedure if your Oracle Business Analytics Warehouse is on a Teradata database and your Informatica Repository is on an MSSQL Server database.

This section contains the following topics:

- [Section D.3.1, "Entering Values for the reset\\_infa\\_seq\\_gen\\_MSSQL.bat File on the Base Informatica Repository"](#)
- [Section D.3.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)
- [Section D.3.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)

### D.3.1 Entering Values for the reset\_infa\_seq\_gen\_MSSQL.bat File on the Base Informatica Repository

Follow this procedure to enter values for the reset\_infa\_seq\_gen\_MSSQL.bat file to initialize the Informatica sequence generator for incremental runs on the base Informatica Repository (for example, Oracle\_BI\_DW\_Base.rep).

1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\Teradata.
2. Open the reset\_infa\_seq\_gen\_MSSQL.bat file.
3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific).

**Table D-5 Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_MSSQL.bat**

| Parameter                                      | Setting                                                                                                                                                                                          |
|------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP_SERVER                                    | Enter the name of the Teradata Server for the OLAP database.                                                                                                                                     |
| OLAP_DATABASE                                  | Enter the name of the database on which OLAP data is available.                                                                                                                                  |
| OLAP_USERNAME                                  | Enter the user ID of the data warehouse database.                                                                                                                                                |
| OLAP_PASSWORD                                  | Enter the password of the data warehouse database.                                                                                                                                               |
| INFORMATICA<br>REPOSITORY_SERVER<br>ADDRESS    | Enter the Informatica Repository Service host address.                                                                                                                                           |
| INFORMATICA<br>REPOSITORY_DB<br>SERVER_ADDRESS | Enter the database address for the Informatica Repository database server in the format <MSSQL Server address>,<port number><br>For example: tcp:192.168.01.2,20100<br>or tcp:MSSQL2005DBA,20100 |
| INFORMATICA<br>REPOSITORY<br>DATABASE          | Name of the Informatica Repository database.                                                                                                                                                     |
| INFORMATICA<br>REPOSITORY_DB<br>USERNAME       | Enter the user ID of the base Informatica Repository database.                                                                                                                                   |

**Table D–5 (Cont.) Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_**

| Parameter                                | Setting                                                                                                                                                                                                                                                                     |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| INFORMATICA<br>REPOSITORY_DB<br>PASSWORD | Enter the password @connection string of the base Informatica Repository database.                                                                                                                                                                                          |
| INFORMATICA<br>REPOSITORY                | Enter the name of the base Informatica Repository.                                                                                                                                                                                                                          |
| INFORMATICA<br>REPOSITORY<br>USERNAME    | Enter the user ID of the base Informatica Repository.                                                                                                                                                                                                                       |
| INFORMATICA<br>REPOSITORY<br>PASSWORD    | Enter the password for the base Informatica Repository.                                                                                                                                                                                                                     |
| INFORMATICA_FOLDER                       | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen_MSSQL.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                         | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                                      |

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.3.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.3.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen\_MSSQL.bat until you have completed the procedures in either [Section D.3.2](#) or [Section D.3.3](#).

### D.3.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.3.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```

- d. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen\_MSSQL.bat.

### D.3.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.3.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.

- c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```
  - d. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:
    - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to:
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
    - b. Add the following lines after the section referred to in step a:
 

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat

IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```
    - c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
    - d. On the data warehouse, run the following SQL statements:
 

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.
    - e. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
  3. Execute reset\_infa\_seq\_gen\_MSSQL.bat.

## D.4 Running reset\_infa\_seq\_gen.bat on Upgrade Informatica Repository

Follow this procedure to run the reset\_infa\_seq\_gen.bat file on the Base informatica Repository if your Oracle Business Analytics Warehouse runs on an Oracle, DB2, or MSSQL Server database.

This section contains the following topics:

- [Section D.4.1, "Entering Values for the reset\\_infa\\_seq\\_gen.bat File on the Upgrade Informatica Repository"](#)
- [Section D.4.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)
- [Section D.4.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)

## D.4.1 Entering Values for the reset\_infa\_seq\_gen.bat File on the Upgrade Informatica Repository

Follow this procedure to run the reset\_infa\_seq\_gen.bat file to initialize the Informatica sequence generator for incremental runs on the Upgrade repository if your Oracle Business Analytics Warehouse runs on an Oracle, DB2, or MSSQL Server database.

1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\<<database type>.
2. Open the reset\_infa\_seq\_gen.bat file.
3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific). See one of the following tables:
  - [Table D-6, " Oracle Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)
  - [Table D-7, " SQL Server Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)
  - [Table D-8, " IBM DB2 Database Parameter Settings for reset\\_infa\\_seq\\_gen.bat"](#)

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.4.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.4.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen.bat until you have completed the procedures in either [Section D.4.2](#) or [Section D.4.3](#).

### Oracle Database Parameter Settings for the Upgrade Informatica Repository

**Table D-6 Oracle Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                       | Setting                                                                              |
|---------------------------------|--------------------------------------------------------------------------------------|
| OLAP USERNAME                   | Enter the data warehouse database user ID.                                           |
| OLAP PASSWORD                   | Enter the user ID/password@connection string of the data warehouse database.         |
| INFORMATICA SERVER ADDRESS      | Enter the Informatica Server host address.                                           |
| INFORMATICA DATABASE TNSENTRY   | Enter the TNS entry for the Informatica Server database.                             |
| INFORMATICA DB USERNAME         | Enter the user ID of the upgrade Informatica Repository database.                    |
| INFORMATICA DB PASSWORD         | Enter the password@connection string of the upgrade Informatica Repository database. |
| INFORMATICA REPOSITORY          | Enter the name of the upgrade Informatica Repository.                                |
| INFORMATICA REPOSITORY USERNAME | Enter the user ID of the upgrade Informatica Repository.                             |
| INFORMATICA REPOSITORY PASSWORD | Enter the password for the upgrade Informatica Repository.                           |



**Table D-6 (Cont.) Oracle Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter          | Setting                                                                |
|--------------------|------------------------------------------------------------------------|
| INFORMATICA_FODLER | Enter the value UPGRADE                                                |
| INFORMATICA_PORT   | Enter the port number for the Repository Service. The default is 6001. |

**SQL Server Database Parameter Settings for the Upgrade Informatica Repository****Table D-7 SQL Server Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                             | Setting                                                                                           |
|---------------------------------------|---------------------------------------------------------------------------------------------------|
| OLAP_SERVER                           | Enter the name of the SQL Server for the OLAP database.                                           |
| OLAP_DATABASE                         | Enter the name of the database on which OLAP data is available.                                   |
| OLAP_USERNAME                         | Enter the user ID of the data warehouse database.                                                 |
| OLAP_PASSWORD                         | Enter the user ID/password@connection string of the data warehouse database.                      |
| INFORMATICA_DB_SERVER_ADDRESS         | Enter the Informatica Server host address.                                                        |
| INFORMATICA_DATABASE                  | Enter the name of the database in which the upgrade Informatica Repository metadata is available. |
| INFORMATICA_DB_USERNAME               | Enter the user ID of the upgrade Informatica Repository database.                                 |
| INFORMATICA_DB_PASSWORD               | Enter the password @connection string of the upgrade Informatica Repository database.             |
| INFORMATICA_REPOSITORY_SERVER_ADDRESS | Enter the name of the machine that runs the Repository Service for the upgrade Repository.        |
| INFORMATICA_REPOSITORY_USERNAME       | Enter the user ID of the upgrade Informatica Repository.                                          |
| INFORMATICA_REPOSITORY_PASSWORD       | Enter the password of the upgrade Informatica Repository.                                         |
| INFORMATICA_FOLDER                    | Enter the value UPGRADE                                                                           |
| INFORMATICA_PORT                      | Enter the port number for the Repository Service. The default is 6001.                            |

**IBM DB2 Database Parameter Settings for the Upgrade Informatica Repository****Table D-8 IBM DB2 Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter           | Setting                                                                               |
|---------------------|---------------------------------------------------------------------------------------|
| OLAP_USERNAME       | Enter the user ID of the data warehouse database.                                     |
| OLAP_PASSWORD       | Enter the user ID/password@connection string of the data warehouse database.          |
| INFA_DATABASE_ALIAS | Enter the name of the database in which Informatica Repository metadata is available. |

**Table D–8 (Cont.) IBM DB2 Database Parameter Settings for reset\_infa\_seq\_gen.bat**

| Parameter                       | Setting                                                                               |
|---------------------------------|---------------------------------------------------------------------------------------|
| INFORMATICA_DB_USERNAME         | Enter the user ID of the upgrade Informatica Repository database.                     |
| INFORMATICA_DB_PASSWORD         | Enter the password @connection string of the upgrade Informatica Repository database. |
| INFORMATICA_REPOSITORY          | Enter the name of the upgrade Informatica Repository.                                 |
| INFORMATICA_REPOSITORY_USERNAME | Enter the user ID of the upgrade Informatica Repository.                              |
| INFORMATICA_REPOSITORY_PASSWORD | Enter the password for the upgrade Informatica Repository.                            |
| INFORMATICA_SERVER_ADDRESS      | Enter the Informatica Server host address.                                            |
| INFORMATICA_FOLDER              | Enter the value UPGRADE                                                               |
| INFORMATICA_PORT                | Enter the port number for the Repository Service. The default is 6001.                |

## D.4.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.4.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```
  - d. In the reset\_infa\_seq\_gen.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:
  - a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to:
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1
>>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2
>>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen.bat.

### D.4.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.4.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_
SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```
  - d. In the reset\_infa\_seq\_gen.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:
  - a. In the reset\_infa\_seq\_gen.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_
SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_of_Wrk_Band -c MAX1
>>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_
PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_of_Wrk_Band -c MAX2
>>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.

- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_
CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.

3. Execute reset\_infa\_seq\_gen.bat.

## D.5 Running reset\_infa\_seq\_gen\_ORA.bat on Upgrade Repository for DW on Teradata and Repository on Oracle

Follow this procedure if your Oracle Business Analytics Warehouse is on a Teradata database and your Informatica Repository is on an Oracle database.

This section contains the following topics:

- [Section D.5.1, "Entering Values for the reset\\_infa\\_seq\\_gen\\_ORA.bat File on the Upgrade Informatica Repository"](#)
- [Section D.5.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)
- [Section D.5.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)

### D.5.1 Entering Values for the reset\_infa\_seq\_gen\_ORA.bat File on the Upgrade Informatica Repository

Follow this procedure to enter values for the reset\_infa\_seq\_gen\_ORA.bat file to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\Teradata.
2. Open the reset\_infa\_seq\_gen\_ORA.bat file.
3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific).

**Table D-9 Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_ORA.bat**

| Parameter                          | Setting                                                                                                                                                                                                                                                                   |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP_SERVER                        | Enter the name of the Teradata Server for the OLAP database.                                                                                                                                                                                                              |
| OLAP_DATABASE                      | Enter the name of the database on which OLAP data is available.                                                                                                                                                                                                           |
| OLAP_USERNAME                      | Enter the user ID of the data warehouse database.                                                                                                                                                                                                                         |
| OLAP_PASSWORD                      | Enter the password of the data warehouse database.                                                                                                                                                                                                                        |
| INFORMATICA_SERVER_ADDRESS         | Enter the Informatica Server host address.                                                                                                                                                                                                                                |
| INFORMATICA_REPOSITORY_DB_USERNAME | Enter the user ID of the Upgrade Informatica Repository database.                                                                                                                                                                                                         |
| INFORMATICA_REPOSITORY_DB_PASSWORD | Enter the password @connection string of the Upgrade Informatica Repository database.                                                                                                                                                                                     |
| INFORMATICA_REPOSITORY             | Enter the name of the Upgrade Informatica Repository.                                                                                                                                                                                                                     |
| INFORMATICA_REPOSITORY_USERNAME    | Enter the user ID of the Upgrade Informatica Repository.                                                                                                                                                                                                                  |
| INFORMATICA_REPOSITORY_PASSWORD    | Enter the password for the Upgrade Informatica Repository.                                                                                                                                                                                                                |
| INFORMATICA_FOLDER                 | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen_ORA.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                   | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                                    |

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.5.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.5.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen\_ORA.bat until you have completed the procedures in either [Section D.5.2](#) or [Section D.5.3](#).

## D.5.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.5.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```
  - d. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:
  - a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to:
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. Add the following lines after the section referred to in step a:
 

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat

IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```
  - c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
  - d. On the data warehouse, run the following SQL statements:
 

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.
  - e. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen\_ORA.bat.

### D.5.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.5.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:

- a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. In the code example, note the value 1234567. You will replace this value in a following step.

- c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```

- d. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the value 1234567 with the value that was returned by the SQL statement.

2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen\_ORA.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prds_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.

- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen\_ORA.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.

3. Execute reset\_infa\_seq\_gen\_ORA.bat.

## D.6 Running reset\_infa\_seq\_gen\_MSSQL.bat on Upgrade Repository for DW on Teradata and Repository on MSSQL Server

Follow this procedure if your Oracle Business Analytics Warehouse is on a Teradata database and your Informatica Repository is on an MSSQL Server database.

This section contains the following topics:

- [Section D.6.1, "Entering Values for the reset\\_infa\\_seq\\_gen\\_MSSQL.bat File on the Upgrade Informatica Repository"](#)
- [Section D.6.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)
- [Section D.6.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#)

### D.6.1 Entering Values for the reset\_infa\_seq\_gen\_MSSQL.bat File on the Upgrade Informatica Repository

Follow this procedure to enter values for the reset\_infa\_seq\_gen\_MSSQL.bat file to initialize the Informatica sequence generator for incremental runs on the Upgrade Informatica Repository.

1. Navigate to OracleBI\dwrep\Upgrade\DbScripts\Teradata.
2. Open the reset\_infa\_seq\_gen\_MSSQL.bat file.
3. Enter the appropriate parameter values to reflect your environment (the parameters are database specific).

**Table D-10 Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_MSSQL.bat**

| Parameter                                      | Setting                                                                                                                                                                                                  |
|------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| OLAP_SERVER                                    | Enter the name of the Teradata Server for the OLAP database.                                                                                                                                             |
| OLAP_DATABASE                                  | Enter the name of the database on which OLAP data is available.                                                                                                                                          |
| OLAP_USERNAME                                  | Enter the user ID of the data warehouse database.                                                                                                                                                        |
| OLAP_PASSWORD                                  | Enter the password of the data warehouse database.                                                                                                                                                       |
| INFORMATICA<br>REPOSITORY_SERVER<br>ADDRESS    | Enter the UPgrade Informatica Repository Service host address.                                                                                                                                           |
| INFORMATICA<br>REPOSITORY_DB<br>SERVER_ADDRESS | Enter the database address for the Upgrade Informatica Repository database server in the format <MSSQL Server address>,<port number><br>For example: tcp:192.168.01.2,20100<br>or tcp:MSSQL2005DBA,20100 |
| INFORMATICA<br>REPOSITORY<br>DATABASE          | Name of the UPgrade Informatica Repository database.                                                                                                                                                     |
| INFORMATICA<br>REPOSITORY_DB<br>USERNAME       | Enter the user ID of the Upgrade Informatica Repository database.                                                                                                                                        |



**Table D-10 (Cont.) Teradata Database Parameter Settings for reset\_infa\_seq\_gen\_**

| Parameter                                | Setting                                                                                                                                                                                                                                                                     |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| INFORMATICA<br>REPOSITORY_DB<br>PASSWORD | Enter the password @connection string of the Upgrade Informatica Repository database.                                                                                                                                                                                       |
| INFORMATICA<br>REPOSITORY                | Enter the name of the Upgrade Informatica Repository.                                                                                                                                                                                                                       |
| INFORMATICA<br>REPOSITORY<br>USERNAME    | Enter the user ID of the Upgrade Informatica Repository.                                                                                                                                                                                                                    |
| INFORMATICA<br>REPOSITORY<br>PASSWORD    | Enter the password for the Upgrade Informatica Repository.                                                                                                                                                                                                                  |
| INFORMATICA_FOLDER                       | For all source systems except Siebel Industry Applications, enter the value HOR.<br><br>For Siebel Industry Applications, you need to run reset_infa_seq_gen_MSSQL.bat twice. For the first execution, enter the value VERT. For the second execution, enter the value HOR. |
| INFORMATICA_PORT                         | Enter the port number for the Repository Service. The default is 6001.                                                                                                                                                                                                      |

After you complete this step, proceed to one of the following sections:

- Follow the steps in [Section D.6.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, or 7.9.5.1, or 7.9.6.
- Follow the steps in [Section D.6.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D"](#) if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

Do not execute reset\_infa\_seq\_gen\_MSSQL.bat until you have completed the procedures in either [Section D.6.2](#) or [Section D.6.3](#).

## D.6.2 Obtaining ROW\_WID from W\_ORG\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_ORG\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3, follow the steps in [Section D.6.3, "Obtaining ROW\\_WID from W\\_PARTY\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.
  - c. On the data warehouse, run the following SQL statement:

```
SELECT MAX(ROW_WID)+1 FROM W_ORG_D
```

- d. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:

- a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to:

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```

- b. Add the following lines after the section referred to in step a:

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat
```

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```

- c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
- d. On the data warehouse, run the following SQL statements:

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.

- e. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen\_MSSQL.bat.

### D.6.3 Obtaining ROW\_WID from W\_PARTY\_D and W\_PRD\_OF\_WRK\_BAND\_D

Follow this procedure to obtain the maximum ROW\_WID value from W\_PARTY\_D and from W\_PRD\_OF\_WRK\_BAND\_D if you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.6.1, 7.9.6.2, or 7.9.6.3.

**Note:** If you are in an upgrade phase in which you are upgrading to Oracle BI Applications version 7.9.0, 7.9.1, 7.9.2, 7.9.3, 7.9.4, 7.9.5, 7.9.5.1, or 7.9.6, follow the steps in [Section D.6.2, "Obtaining ROW\\_WID from W\\_ORG\\_D and W\\_PRD\\_OF\\_WRK\\_BAND\\_D."](#)

1. Obtain the maximum ROW\_WID value from W\_ORG\_D:
  - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. In the code example, note the value 1234567. You will replace this value in a following step.

- c. On the data warehouse, run the following SQL statement:
 

```
SELECT MAX(ROW_WID)+1 FROM W_PARTY_D
```
- d. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the value 1234567 with the value that was returned by the SQL statement.
2. Obtain the maximum ROW\_WID value from W\_PRD\_OF\_WRK\_BAND\_D:
  - a. In the reset\_infa\_seq\_gen\_MSSQL.bat file, go to the end of the file and locate the section similar to:
 

```
IF %INFA_REP%==UPGRADE echo pmrep Updateseqgenvals -f"UPGRADE_7951_to_796_SBL" -t Seq_W_PARTY_D_Wid -c 1234567 >>sequence_gen_update.bat.
```
  - b. Add the following lines after the section referred to in step a:
 

```
IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_3000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX1 >>sequence_gen_update.bat

IF %INFA_FOLDER%==HOR echo pmrep Updateseqgenvals -f "SILOS" -m SIL_PeriodOfWorkBandDimension -t Seq_4000_Plus_Sequence_Prđ_Of_Wrk_Band -c MAX2 >>sequence_gen_update.bat
```
  - c. In the code example, note the values MAX1 and MAX2. You will replace these values in a following step.
  - d. On the data warehouse, run the following SQL statements:
 

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_EMP'
```

The value returned from the SQL statement above corresponds to MAX1.

```
select max(ROW_WID)+1 from W_PRD_OF_WRK_BAND_D where level_id = 'POW_BAND_CWK'
```

The value returned from the SQL statement above corresponds to MAX2.
  - e. In the reset\_infa\_seq\_gen\_MSSQL.bat file, replace the values MAX1 and MAX2 with the values returned by the SQL statements.
3. Execute reset\_infa\_seq\_gen\_MSSQL.bat.



---

---

## Using the DDLimp Utility

This appendix contains instructions for using the DDLimp utility. It contains the following topic:

- [Section E.1, "Using the DDLimp Utility to Run CTL Files"](#)

### E.1 Using the DDLimp Utility to Run CTL Files

Follow this procedure to use the DDLimp utility to run CTL files.

#### To use the DDLimp utility

1. Navigate to the folder OracleBI\dwrep\Upgrade\CTLFiles, and locate the specified CTL file.
2. If you changed any of the preconfigured column definitions in the CTL file for your implementation, you need to edit the CTL file to reflect the change. For example, if you renamed or resized an existing column in W\_DAY\_D, you need to make the same change in the CTL file. If you do not make the change in the CTL file, when the CTL file is run, the column definition will revert to the preconfigured definition. (As a customization, if a column size was changed from VARCHAR(50) to VARCHAR(100), and the same change was not made in the CTL file before running, the column will revert to the preconfigured value of VARCHAR(50), which could cause data to be truncated in some databases.)
3. For non-Teradata databases, use the DDLimp utility to run the <file\_name>.ctl script. Use the following command:

```
..\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC connect string>
/G SSE_ROLE /I N /R Y /F <..\OracleBI\dwrep\Upgrade\CTLFiles\<FILE_NAME>.CTL>
/L <..\OracleBI\dwrep\<FILE_NAME>.log>
```

For example:

```
DDLIMP /U SADMIN /P SADMIN /C OBIA /G SSE_ROLE
/I N /R Y /F C:\OracleBI\dwrep\Upgrade\CTLFiles\UPGRADE_7962.CTL /L
C:\OracleBI\dwrep\UPGRADE.log
```

4. For Teradata databases, use the DDLimp utility to run the <file\_name>.ctl script. Use the following command:

```
..\OracleBI\dwrep\bin\DDLIMP /U <USER> /P <PASSWORD> /C <ODBC connect string>
/G SSE_ROLE /F <..\OracleBI\dwrep\Upgrade\CTLFiles\<FILE_NAME>.CTL>
/s N /L <..\OracleBI\dwrep\<FILE_NAME>.log>
```

For example:

```
DDLIMP /U SADMIN /P SADMIN /C OBIA /G SSE_ROLE
/F C:\OracleBI\dwrep\Upgrade\CTLFiles\UPGRADE_7962.CTL /s N /L
C:\OracleBI\dwrep\UPGRADE.log
```

## E.2 Additional DDLimp Commands

The following DDLimp commands are also available:

- /P <PASSWORD> - The password for the data warehouse.
- For Oracle databases, use the Data Direct drivers.
- /C <ODBC connect string> - The name of the ODBC connect string.
- /I N - Tells DDLimp to ignore the indexes DDL defined in the CTL file if any exist. It does not change existing indexes. (The default is /I Y, which tells DDLimp to create and merge indexes from the CTL file with the indexes in the database.
- In addition, you can use the following commands:
  - /W Y - If the OLAP database is Oracle and Unicode.
  - /Z Y - If the OLAP database is DB2 or SQL Server and Unicode.
  - /B <TABLE\_SPACE\_NAME> - If you want to create these tables in a separate table space. For DB2, This must be specified as 32K tablespace.
  - /X <INDEX\_TABLE\_SPACE\_NAME> - If you want to create the indexes in a separate tablespace. For DB2, This must be specified as 32K tablespace.
  - /Y - Storage File for DB2/390.
  - /R - Regrant tables.

# Upgrading the Oracle BI Repository for Industry-Specific Analytics Applications

This appendix contains instructions for upgrading the Oracle BI repository for Pharma Analytics, Consumer Sector, and Vehicle Sales.

In Oracle BI Applications release 7.9.4, the Pharma Analytics business model and Core business model were merged. In Oracle BI Applications release 7.9.5, the Consumer Sector and Vehicle Sales Analytics business models and Core business model were merged. Because of this merge, the process of upgrading the Oracle BI repository has some steps that differ from the standard Oracle BI repository upgrade process.

This section includes the following topics:

- [Section F.1, "Common Dimensions"](#)
- [Section F.2, "Merging Siebel Analytics and Oracle BI Repositories"](#)
- [Section F.3, "Replacing Common Dimensions After the Repository Merge"](#)

## F.1 Common Dimensions

[Table F-1](#), [Table F-2](#), and [Table F-3](#) list the common dimensions of Pharma, Consumer Sector, and Vehicle Sales Analytics and their statuses. Some of the dimensions are not shared with other Core modules because of specific requirements.

**Table F-1 Common Pharma Analytics Dimensions**

| Dimensions in Pre-7.9.4 Releases | Dimensions in Release 7.9.4    | Status     | Comments                              |
|----------------------------------|--------------------------------|------------|---------------------------------------|
| Dim - Accounts                   | Dim - Customer                 | Shared     | None                                  |
| Dim - Contacts                   | Dim - Contact                  | Shared     | New name is singular                  |
| Dim - Security Dimension         | Dim - Position Security        | Shared     | None                                  |
| Dim - Time Period                | Dim - Date                     | Shared     | None                                  |
| Dim - Geography                  | Dim - Pharma Geography         | Not shared | Use Pharma-specific dimension in Core |
| Dim - Geography_Account          | Dim - Pharma Geography_Account | Not shared | Use Pharma-specific dimension in Core |
| Dim - Geography_Contact          | Dim - Pharma Geography_Contact | Not shared | Use Pharma-specific dimension in Core |
| Dim - Products                   | Dim - Pharma Products          | Not shared | Use Pharma-specific dimension in Core |

**Table F-2 Common Consumer Sector Analytics Dimensions**

| <b>Dimensions in Pre-7.9.5 Releases</b> | <b>Dimensions in Release 7.9.5</b>  | <b>Status</b> | <b>Comments</b>                                                       |
|-----------------------------------------|-------------------------------------|---------------|-----------------------------------------------------------------------|
| Dim - Account Geography                 | Dim - Account Geography             | Shared        | None                                                                  |
| Dim - Accounts                          | Dim - Customer                      | Shared        | None                                                                  |
| Dim - Accounts Hierarchy                | Dim - Accounts Hierarchy            | Shared        | None                                                                  |
| Dim - Employees                         | Dim - Employees                     | Shared        | None                                                                  |
| Dim - End Date                          | Dim - End Date                      | Shared        | None                                                                  |
| Dim - Position Hierarchies              | Dim - Position                      | Shared        | Dim - Position in 7.9.5 combines both Position and Position Hierarchy |
| Dim - Positions                         | Dim - Position                      | Shared        | Dim - Position in 7.9.5 combines both Position and Position Hierarchy |
| Dim - Product Categories Hierarchy      | Dim - CS Product Category Hierarchy | Not shared    | Use CS-specific dimension in Core                                     |
| Dim - Products                          | Dim - CS Product                    | Not shared    | Use CS-specific dimension in Core                                     |
| Dim - Start Date/Date                   | Dim - Start Date                    | Shared        | None                                                                  |

**Table F-3 Common Vehicle Sales Analytics Dimensions**

| <b>Dimensions in Pre-7.9.5 Releases</b> | <b>Dimensions in Release 7.9.5</b> | <b>Status</b> | <b>Comments</b> |
|-----------------------------------------|------------------------------------|---------------|-----------------|
| Dim - Accounts                          | Dim - Customer                     | Shared        | None            |
| Dim - Accounts Geography                | Dim - Account Geography            | Shared        | None            |
| Dim - Contacts Geography                | Dim - Person Geography             | Shared        | None            |
| Dim - Date                              | Dim - Date                         | Shared        | None            |
| Dim - Households                        | Dim - Households                   | Shared        | None            |
| Dim - Individuals                       | Dim - Contact                      | Shared        | None            |
| Dim - Lease/Loan Expiry                 | Dim - End Date                     | Shared        | None            |
| Dim - Product Hierarchy                 | Dim - Product Hierarchy            | Shared        | None            |
| Dim - Products                          | Dim - Product                      | Shared        | None            |
| Dim - Vehicle                           | Dim - Asset                        | Shared        | None            |

## F.2 Merging Siebel Analytics and Oracle BI Repositories

The Pharma, Consumer Sector, and Vehicle Sales Analytics upgrade process involves merging customizations of prior releases of the Siebel Analytics or Oracle BI

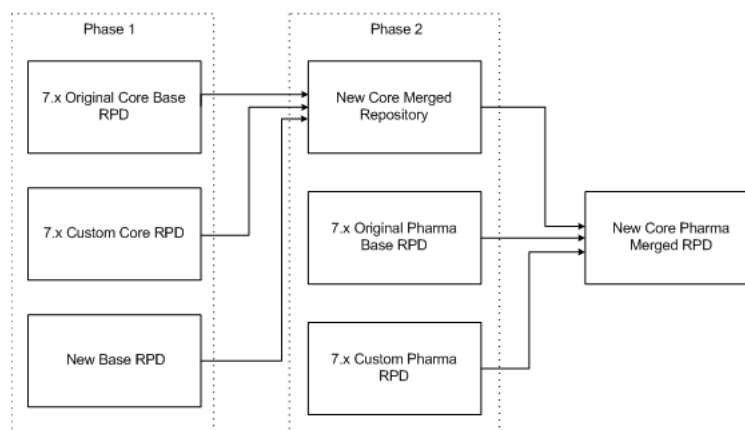


repositories into the current version of the Oracle BI repository. This process follows the same principle as the Core upgrade process but includes some additional steps.

In this section, the upgrade process for Pharma Analytics from pre-7.9.4 releases to release 7.9.4 and higher will be used as an example. The upgrade process for Consumer Sector and Vehicle Sales Analytics from pre-7.9.5 releases to release 7.9.5 and higher are the same as that for the upgrade to Pharma Analytics release 7.9.4 and higher.

As shown in [Figure F-1](#), there are two phases to the Pharma Analytics upgrade process.

**Figure F-1 Phases of Repository Merge for Pharma Analytics**



### Phase 1

As shown in [Figure F-1](#), there are two phases in the Pharma Analytics upgrade process. Phase 1 involves merging the 7.x original Core base repository (the repository that shipped with the release of Siebel Analytics or Oracle BI you are currently running), the 7.x custom Core repository, and the new base repository into one output repository.

### Phase 2

Phase 2 involves merging the output repository of Phase 1 with the 7.x original Pharma Analytics base repository and the 7.x custom Pharma Analytics repository. The output repository of Phase 2 is a merged Pharma and Core repository that includes your customizations from prior releases and the new data model.

---

**Note:** You need to perform both phases of the upgrade process if your current, customized repository has both Core and Pharma models and the Core model is customized.

If your current, customized repository does not include the Core model or includes the Core model but the Core model is not customized, you only need to perform Phase 2.

If you only perform Phase 2, then you will not have a new Core merged RPD (as shown in [Figure F-1](#)). Instead, you will use the new base RPD in place of the new Core merged RPD.

See [Table F-4](#) and [Table F-5](#) for a description of the different RPD file names.

---

The tasks in this section refer to multiple versions of the Siebel Analytics or Oracle BI repository. [Table F-4](#) provides the names and descriptions of the repositories used in Phase 1. [Table F-5](#) provides the names and descriptions of the repositories used in Phase 2.

**Table F-4 Phase 1 Repository Files**

| Repository File           | Description                                                                                                   |
|---------------------------|---------------------------------------------------------------------------------------------------------------|
| 7x_original_Core_base.rpd | The trimmed, original (standard) repository for the version you are upgrading from.                           |
| 7x_custom_Core.rpd        | The trimmed, custom 7.x repository that includes the Core model customizations.                               |
| New_base.rpd              | The trimmed, new repository that includes related Core and Pharma modules in one Core model (logical folder). |

**Table F-5 Phase 2 Repository Files**

| Repository File             | Description                                                                                                                          |
|-----------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| New_Core_merged.rpd         | The output repository from Phase 1.<br>If you do not perform Phase 1, then use the new base RPD in place of the new Core merged RPD. |
| 7x_original_Pharma_base.rpd | The trimmed, original 7.x repository that includes the Pharma model, which was not customized.                                       |
| 7x_custom_Pharma.rpd        | The trimmed, customized repository that includes a customized Pharma model.                                                          |
| New_Core_Pharma_merged.rpd  | The final, merged new repository that includes a single Core model (logical folder), including all the customized content.           |

## F.2.1 Creating Working Folders

You will use the working folders to hold the repository files after the different stages of the merge process.

Create a folder for the merge process, such as \OracleBIPharmaUpgrade, and then create the following subfolders:

- Original
- AfterTrimDown
- AfterEqualize
- AfterMerge
- AfterManualWork
- AfterRegressions

## F.2.2 Trimming Repository Files

Trimming the repositories so that you upgrade only the content that is in use can narrow the upgrade scope and eliminate unnecessary complexities during the merge process.

You can trim down repository files by extracting projects or manually trimming down repository objects.

---

---

**Note:** The Security Group property controls the presentation catalogs that will be extracted. If a presentation catalog is added to a project for extract but none of the groups in the project has visibility to see the catalog, it will not show up in the extracted repository file.

---

---

### F.2.2.1 Trimming the Original Base Repository Files

You should trim the original base repository file to meet your business requirements.

#### To trim the original base repository files

1. If you are performing Phase 1 of the upgrade process, trim the original Core base repository file.
2. Save the file as 7x\_original\_Core\_base.rpd (where 7x represents the release of Siebel Analytics or Oracle BI) in the AfterTrimDown subfolder.
3. Trim the original Pharma base repository file.
4. Save the file as 7x\_original\_Pharma\_base.rpd in the AfterTrimDown subfolder.

### F.2.2.2 Trimming the New Base Repository File

The new base repository file that you received with the current Oracle BI Applications release, contains one Core model (logical folder), which holds Pharma and Core repository objects.

#### To trim the new base repository file

1. Trim the new base repository file.
2. Save the file as New\_base.rpd in the AfterTrimDown subfolder.

### F.2.2.3 Trimming the Custom Repository Files

If you are performing Phase 1 of the upgrade process, you will have a 7x custom Core repository file (7x\_custom\_core.rpd) that contains the 7x original Core base contents and your customizations.

You will also have a 7x custom Pharma repository file (7x\_custom\_Pharma.rpd) that contains the 7x original Pharma base contents and your customizations.

Save these files in the AfterTrimDown subfolder.

## F.2.3 Renaming Objects in the Original Pharma Base Repository File

In the 7x original Pharma base repository file and the 7x custom Pharma repository file, you need to rename some repository objects to be compatible with the new data model.

#### To rename objects in the Pharma repository files

1. In the Server Administration Tool, open the 7x\_original\_Pharma\_base.rpd file.
2. In the Business Model and Mapping layer, rename Business Model Pharma to Core.
3. In the Physical layer, do the following:
  - a. Rename Database Pharma Data Warehouse to Oracle Data Warehouse.
  - b. Make sure the Connection Pool is named as Oracle Data Warehouse Connection Pool.

- c. Make sure the Catalog entry (below the Connection Pool entry) is named as Catalog.
  - d. Make sure the Schema entry (below the Catalog entry) is named as dbo.
4. Repeat Steps 1 through 3 using the 7x\_custom\_Pharma.rpd file.

## F.2.4 Equalizing the Oracle BI Repositories

The equalization process in the standard Oracle BI repository upgrade uses the original base repository from a previous release as a starting point. This type of equalization is referred to as "backward" equalization. The Pharma upgrade uses what is called "forward" equalization, in which you use the repository file from the current release as a starting point and equalize the 7x base repository file and the 7x custom repository file to it.

You will first need to prepare the MAP file for "forward" equalization before you execute the equalization.

Table F-6 provides a list of the available MAP files and the Siebel Analytics or Oracle BI Applications release version associated with the file.

**Table F-6 Rename MAP Files to Be Used for Various Releases**

| Siebel Analytics / Oracle Business Intelligence Applications Release Version (Upgrading from DW Version) | Rename MAP File to Be Used |
|----------------------------------------------------------------------------------------------------------|----------------------------|
| Siebel Business Analytics Applications 7.0.x                                                             | Not available              |
| Siebel Business Analytics Applications 7.5.x                                                             | Not available              |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP Pre-7.7.0)                            | Rename77-7963.map          |
| Siebel Business Analytics Applications 7.7.x (with Siebel CRM OLTP 7.7.0)                                | Rename771-7963.map         |
| Siebel Business Analytics Applications 7.8.2 and all 7.8.x versions before this release                  | Rename782-7963.map         |
| Siebel Business Analytics Applications 7.8.3 and all 7.8.x versions after this release                   | Rename783-7963.map         |
| Oracle BI Applications 7.9.0                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.1                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.2                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.3                                                                             | Rename793to7963.map        |
| Oracle BI Applications 7.9.4                                                                             | Rename794to7963.map        |
| Oracle BI Applications 7.9.5                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.5.1                                                                           | Rename7951to7963.map       |
| Oracle BI Applications 7.9.5.2                                                                           | Rename7951to7963.map       |
| Oracle BI Applications 7.9.6                                                                             | Rename79x-7963.map         |
| Oracle BI Applications 7.9.6.2                                                                           | Not available              |

### To prepare the MAP file for "forward" equalization

1. Copy the appropriate MAP file from the \OracleBI\Upgrade folder into the folder where you will execute equalizerpds.exe.

---



---

**Note:** The naming convention of the MAP files is rename<release from which you are upgrading>-<release to which you are upgrading>.map.

For example if you are upgrading from release 7.8.4 to release 7.9.5, you should use the file rename784\_795.map.

---



---

2. Open the MAP file in Excel.
3. In the Text Import Wizard, accept the default in the Step 1 dialog by clicking Next.
4. In the Step 2 dialog of the Text Import Wizard, do the following:
  - a. Make sure the Tab check box is selected in the Delimiters region.
  - b. In the Text qualifier list, select None.
  - c. Click Next.
5. In the Step 3 dialog, click Finish.  
The file opens in an Excel spreadsheet window and contains three columns.
6. Switch the order of the second and third columns by cutting the third column and inserting it as the second column.
7. Save the file.
8. Check to make sure the string format is correct.

#### To equalize the repository files

1. Run equalizerpds.exe.

An example of the equalizerpds command is as follows:

```
equalizerpds -A Administrator -B SADMIN
-C \OracleBIPharmaUpgrade\AfterTrimDown\New_base.rpd
-D Administrator -E SADMIN
-F \OracleBIPharmaUpgrade\AfterTrimDown\7x_custom_Pharma.rpd
-O \OracleBIPharmaUpgrade\AfterEqualize\7x_custom_Pharma.rpd
-X -J rename784-795.map
```

If the equalizerpds.exe executable file runs correctly, no errors are returned.

2. Repeat Step 1 to equalize each of the files in AfterTrimDown to New\_base.rpd.
3. To verify the process completed successfully, compare the size of the repositories. The output repository (-O) should be close to the same size as the repository you equalized (-F).

## F.2.5 Merging the Core Repository Files

---



---

**Note:** You need to perform this procedure if your customized repository has both Core and Pharma models and the Core model is customized. If the customized repository does not include the Core model or includes the Core model but the Core model is not customized, you do not need to perform this step.

---



---

This step involves a three-way merge on the Core repository. For a description of the files that will be merged in this procedure, see [Section F.2, "Merging Siebel Analytics and Oracle BI Repositories."](#)

**To merge the core repository files**

1. Copy 7x\_original\_Core\_base.rpd, 7x\_custom\_Core.rpd, and New\_base.rpd to the AfterMerge folder.
2. In the Server Administration Tool, open the New\_base.rpd file.
3. From the Administration Tool menu bar, select File, then select Merge.
4. In the Select Original Repository dialog, select 7x\_original\_Core\_base.
5. Enter the password, and click OK.
6. Click Select for the Modified Repository field.
7. In the Select Modified Repository dialog, select 7x\_custom\_Core.rpd.
8. Click Open, type the password, and then click OK.
9. In the Decision list, select the action you want to take regarding the repository change, or accept the default action.

For information about making decisions, see [Section F.2.7, "Making Merge Decisions in the Administration Tool."](#)

10. To locate subsequent rows with empty Decision fields, click the Decision header cell.

When all rows have a value in the Decision field, the Merge button is enabled.

11. Click Merge.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

12. Click Yes when asked if you want to run a consistency check.

The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300 you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

13. Save the merged repository as New\_Core\_merged.rpd .

## F.2.6 Merging the Pharma Repository Files

This step involves a three-way merge on the Pharma repository. For a description of the files that will be merged in this procedure, see [Section F.2, "Merging Siebel Analytics and Oracle BI Repositories."](#)

---



---

**Note:** If you did not perform Phase 1 of the upgrade process you will not have a New\_Core\_merged.rpd file. In place of the New\_Core\_merged.rpd file, you should use the New\_new\_base.rpd file.

---



---

### To merge the Pharma repository files

1. Copy New\_Core\_merged.rpd, 7x\_original\_Pharma\_base.rpd, and 7x\_custom\_Pharma.rpd to the AfterMerge folder.
2. In the Server Administration Tool, open the New\_Core\_merged.rpd file.
3. From the Administration Tool menu bar, select File, then select Merge.
4. In the Select Original Repository dialog, select 7x\_original\_Pharma\_base.
5. Enter the password, and click OK.
6. Click Select for the Modified Repository field.
7. In the Select Modified Repository dialog, select 7x\_custom\_Pharma.rpd.
8. Click Open, type the password, and then click OK.
9. In the Decision list, select the action you want to take regarding the repository change, or accept the default action.

For information about making decisions, see [Section F.2.7, "Making Merge Decisions in the Administration Tool."](#)

10. To locate subsequent rows with empty Decision fields, click the Decision header cell.

When all rows have a value in the Decision field, the Merge button is enabled.

11. Click Merge.

This process can take up to 40 minutes, depending on the size of the repositories you are working with. A message will alert you when the merge is complete.

12. Click Yes when asked if you want to run a consistency check.

The number of errors returned by the consistency check is an indication of how successful the merge process was. If you receive many errors, for example, over 300, you should analyze the reason for the errors. If the merge process failed to recognize that two objects are the same, you may need to edit the rename file or add your own rename file if you have renamed many of the objects and the upgrade engine failed to relate them to the original objects.

You also may need to change the actions you selected in the Decision drop-down list before rerunning the merge. This could save you time by reducing the number of errors that you will need to fix manually.

Once you are satisfied with the results of the merge, you should fix the remaining errors manually. It is important that you fix all errors before moving on to the next step. This repository serves as the input for the next stage.

You should also check that all of your customized objects are present and that no duplicate physical tables were introduced. To check for duplicate tables, search for physical tables using a query such as:

```
where name like '*#1'
```

13. Save the merged repository as `New_Core_Pharma_merged.rpd`.

## F.2.7 Making Merge Decisions in the Administration Tool

When making decisions about merging repository objects in the Administration Tool, you should consider the following points:

- For objects that do not appear in the new base repository, you should normally choose "current," which will incorporate the changes into the new base repository.
- For objects added to the new base repository, you should normally choose "current," which will keep the changes.
- For objects in the original repository that were replaced with new objects, you may see decisions for removing the old objects from the current repository and adding the new objects. Choosing "current" will replace the old objects.
- For new customizations that you added to a repository, choose "modified" to keep the changes.
- If an object is changed in both the customized repository and the new base repository, the description "changed in both," may appear. In such cases, choose "current" to keep the object as it is in the new base repository, or choose "modified," to keep the object as it is in the customized repository.

## F.3 Replacing Common Dimensions After the Repository Merge

After the repository file merge, all preconfigured presentation dimension tables and columns should be merged properly and sourcing from the new logical dimensions in the Core model. For example, Account Name in the Account presentation table should source from "Core"."Dim - Customer"."Account Name," and Gross Margin in the Product presentation table should source from "Core"."Dim - Pharma Products"."Gross Margin."

However, customized presentation tables and columns may still source from old dimensions. [Table F-7](#), [Table F-8](#), and [Table F-9](#) list the common dimensions that have new names in the new release. If you customized presentation tables or columns from the old dimensions listed in [Table F-7](#), [Table F-8](#), and [Table F-9](#) you need to replace the old logical source tables or table columns with the new ones.

A way to quickly allocate these problematic presentation tables or columns would be to do the following:

1. In the Core logical folder in the Server Administration Tool, find the old dimension name as listed in [Table F-7](#), [Table F-8](#), or [Table F-9](#), for example, "Dim - Accounts."
2. Right-click the old dimension name, then click Display Related, and then click Presentation Column.
3. Replace the presentation columns with the same columns from the new logical dimension, for example, "Dim - Customer"



4. To verify you replaced the presentation columns correctly, search for presentation columns on the Core logical dimension, such as "Core - Accounts". If the return is empty, then it is safe to delete the old dimension, for example, "Dim - Accounts"

The following common dimensions in 7.x need to be reviewed and removed.

**Table F-7 Names for Pharma Analytics Common Dimensions in Pre-7.9.4 Releases**

| Name of Dimension in Pre-7.9.4 Releases | Name of Dimension in Release 7.9.4 | Status | Comments             |
|-----------------------------------------|------------------------------------|--------|----------------------|
| Dim - Accounts                          | Dim - Customer                     | Shared | None                 |
| Dim - Contacts                          | Dim - Contact                      | Shared | New name is singular |
| Dim - Time Period                       | Dim - Date                         | Shared | None                 |

**Table F-8 Names for Consumer Sector Analytics Common Dimensions in Pre-7.9.5 Releases**

| Name of Dimension in Pre-7.9.5 Releases | Name of Dimension in Release 7.9.5 | Status | Comments                                                     |
|-----------------------------------------|------------------------------------|--------|--------------------------------------------------------------|
| Dim - Accounts                          | Dim - Customer                     | Shared | None                                                         |
| Dim - Position Hierarchies              | Not applicable                     | Shared | Merged into Dim - Position in Core                           |
| Dim - Positions                         | Dim - Position                     | Shared | Dim - Position combines both Position and Position Hierarchy |
| Dim - Start Date/Date                   | Dim - Start Date                   | Shared | None                                                         |

**Table F-9 Names for Vehicle Sales Analytics Common Dimensions in Pre-7.9.5 Releases**

| Name of Dimension in Pre-7.9.5 Releases | Name of Dimension in Release 7.9.5 | Status | Comments |
|-----------------------------------------|------------------------------------|--------|----------|
| Dim - Accounts                          | Dim - Customer                     | Shared | None     |
| Dim - Accounts Geography                | Dim - Account Geography            | Shared | None     |
| Dim - Contacts Geography                | Dim - Person Geography             | Shared | None     |
| Dim - Individuals                       | Dim - Contact                      | Shared | None     |
| Dim - Lease/Loan Expiry                 | Dim - End Date                     | Shared | None     |
| Dim - Products                          | Dim - Product                      | Shared | None     |
| Dim - Vehicle                           | Dim - Asset                        | Shared | None     |



---

---

# Index

## Symbols

---

- \$\$Hint1 parameter for Oracle databases
  - configuring, A-12
- \$\$PERIOD parameter
  - configuring for Oracle Supply Chain and Order Management Analytics, A-13

## A

---

- Application-specific parameters for Oracle Financial Analytics
  - configuring in 796\_UPG\_PARAMS.txt, A-10

## B

---

- best practices
  - preparing to upgrade, 2-5

## C

---

- COGS fact mapping
  - configuring 7962\_UPG\_PARAMS.txt for Oracle EBS R12, A-15
- COGS fact mapping for Oracle EBS R12
  - configuring 796\_UPG\_PARAMS.txt, A-11
- Consumer Sector
  - upgrading Oracle BI repository, F-1

## D

---

- DAC
  - installing for Siebel Analytics 7.5.x, 3-5
  - upgrading and configuring for Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-5
  - upgrading for JD Edwards source systems, 8-4
  - upgrading for Oracle EBS source systems, 6-4
  - upgrading for PeopleSoft source systems, 7-4
  - upgrading for Siebel sources, 5-4
- data warehouse schema
  - upgrading for JD Edwards source systems, 8-6
  - upgrading for Oracle EBS source systems, 6-6
  - upgrading for PeopleSoft source systems, 7-6
  - upgrading for Siebel Analytics 7.5.x, 3-5
  - upgrading for Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-9
  - upgrading for Siebel Business Analytics on

- Teradata database, 4-28
- upgrading for Siebel sources, 5-7
- DDLimp utility
  - using to run CTL files, E-1

## F

---

- FSG Hierarchies, A-10

## G

---

- GL Data Extraction
  - configuring parameter in 796\_UPG\_PARAMS.txt, A-10
  - configuring parameter in 7962\_UPG\_PARAMS.txt for PeopleSoft sources, A-14

## I

---

- indexes
  - generating and running scripts for dropping and creating, B-1
- Informatica
  - parameter files, configuring, A-1
  - upgrading for JD Edwards source systems, 8-2
  - upgrading for Oracle EBS, 6-2
  - upgrading for PeopleSoft source systems, 7-2
  - upgrading for Siebel Analytics 7.5.x, 3-3
  - upgrading for Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-3
  - upgrading for Siebel sources, 5-2
- Informatica parameter file 793\_UPG\_PARAMS.txt
  - configuring, A-1
- Informatica parameter file 794\_UPG\_PARAMS.txt
  - configuring, A-2
- Informatica parameter file 795\_UPG\_PARAMS.txt
  - configuring, A-2
- Informatica parameter file 7951\_UPG\_PARAMS.txt
  - configuring, A-3
- Informatica parameter file 796\_UPG\_PARAMS.txt
  - configuring, A-3, A-7
- Informatica parameter file 7961\_UPG\_PARAMS.txt
  - configuring, A-4
- Informatica parameter file 7962\_UPG\_PARAMS.txt
  - configuring, A-5
- Informatica parameter file 7963\_UPG\_PARAMS.txt

- configuring, A-6
- Informatica Repository
  - upgrading for Siebel Analytics 7.5.x, 3-3
- Informatica sequence generator
  - setting parameters, D-1

## J

---

- JD Edwards
  - upgrading, 8-1

## L

---

- LDIF file
  - importing into embedded LDAP Server for JD Edwards sources, 8-20
  - importing into embedded LDAP Server for Oracle EBS sources, 6-65
  - importing into embedded LDAP Server for PeopleSoft sources, 7-53
  - importing into embedded LDAP Server for Siebel Analytics 7.5.x, 3-26
  - importing into embedded LDAP Server for Siebel Business Analytics, 4-56
  - importing into embedded LDAP Server for Siebel sources, 5-48

## O

---

- Oracle BI Applications
  - upgrading for JD Edwards source systems, 8-2
  - upgrading for Oracle EBS source systems, 6-2
  - upgrading for PeopleSoft source systems, 7-2
  - upgrading for Siebel Analytics 7.5.x, 3-2
  - upgrading for Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-3
  - upgrading for Siebel sources, 5-2
- Oracle BI Enterprise Edition
  - considerations for upgrading to 11g, 2-3
- Oracle BI Infrastructure
  - upgrading for JD Edwards source systems, 8-2
  - upgrading for Oracle BI Applications for Siebel sources, 5-2
  - upgrading for Oracle EBS source systems, 6-2
  - upgrading for PeopleSoft source systems, 7-2
  - upgrading for Siebel Analytics 7.5.x, 3-2
  - upgrading for Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-2
- Oracle BI Repository
  - merging for Oracle EBS sources, 6-67
  - merging for Siebel Analytics 7.5.x, 3-27
  - merging for Siebel Business Analytics, 4-58
  - merging for Siebel sources, 5-49
  - upgrading for JD Edwards sources, 8-20
  - upgrading for Oracle EBS sources, 6-66
  - upgrading for PeopleSoft sources, 7-53
  - upgrading for Siebel sources, 5-48
- Oracle EBS
  - upgrading, 7-1
- Oracle Financial Analytics
  - configuring GL Data Extraction in 796\_UPG\_

- PARAMS.txt, A-10
- configuring parameters, A-10
- creating indexes for, C-1
- FSG Hierarchies, A-10
- setting parameters, A-13
- setting parameters for Oracle BI Applications release 7.9.6.3, A-15
- Value Set Hierarchies, A-10
- Oracle Human Resources Analytics
  - setting parameters for Oracle BI Applications release 7.9.6.3, A-17
- Oracle Procurement and Spend Analytics
  - setting parameters for Oracle BI Applications release 7.9.6.3, A-17, A-18
- Oracle Project Analytics
  - configuring parameters in 796\_UPG\_PARAMS.txt, A-12
- Oracle Supply Chain and Order Management Analytics
  - configuring \$\$PERIOD parameter, A-13
  - configuring TIME\_GRAIN parameter, A-13
  - setting parameters, A-13

## P

---

- PeopleSoft
  - upgrading, 7-1
- PeopleSoft source systems
  - upgrading, 7-1
- Pharma Analytics
  - upgrading Oracle BI repository, F-1
- policy store
  - localizing for JD Edwards sources, 8-20
  - localizing for Oracle EBS sources, 6-65
  - localizing for PeopleSoft sources, 7-52
  - localizing for Siebel Analytics 7.5.x, 3-26
  - localizing for Siebel Business Analytics, 4-56
  - localizing for Siebel sources, 5-47
- presentation catalog
  - merging for JD Edwards sources, 8-21
  - merging for Oracle EBS sources, 6-72
  - merging for PeopleSoft sources, 7-54
  - merging for Siebel Analytics 7.5.x, 3-32
  - merging for Siebel Business Analytics, 4-63
  - merging for Siebel sources, 5-54, 5-55
  - upgrading for JD Edwards sources, 8-20
  - upgrading for Oracle EBS sources, 6-66
  - upgrading for PeopleSoft sources, 7-53
  - upgrading for Siebel sources, 5-48
- preupgrade
  - best practices, 2-5
- process flow, 2-6

## R

---

- reset\_infa\_seq\_gen.bat
  - setting database-specific parameters, D-1

## S

---

- Siebel Analytics

- upgrading release 7.5.x, 3-1
- Siebel Analytics Repository
  - upgrading for Siebel Analytics 7.5.x, 3-26
- Siebel Analytics Web Catalog
  - upgrading for Siebel Analytics 7.5.x, 3-26
- Siebel Business Analytics
  - upgrading release 7.7, 7.7.x, 7.8.x, 4-1
  - upgrading releases 7.7, 7.7.x and 7.8.x, 4-1
- Siebel Business Analytics Repository
  - upgrading for Siebel Business Analytics, 4-57
- Siebel Business Analytics Web Catalog
  - upgrading for Siebel Business Analytics, 4-57
- stages of upgrade
  - process flow, 2-6
  - sequence of stages, 2-6
- supported upgrade paths, 2-1

## T

---

- Teradata database
  - upgrading data warehouse schema for Oracle EBS source systems, 6-23
  - upgrading data warehouse schema for PeopleSoft source systems, 7-21
  - upgrading data warehouse schema for Siebel Analytics 7.7, 7.7.x, 7.8.x, 4-28
  - upgrading data warehouse schema for Siebel sources, 5-20
- TIME\_GRAIN parameter
  - configuring for Oracle Supply Chain and Order Management Analytics, A-13

## U

---

- upgrade
  - best practices, 2-5
  - considerations for upgrading Oracle BI Applications, 2-3
  - considerations for upgrading to Oracle BI Enterprise Edition 11g, 2-3
  - major stages, 2-6
  - overview, 2-1
  - preparing, 2-5
  - preupgrade considerations, 2-3
  - process flow, 2-6
  - reasons to upgrade, 2-2
  - sequence of steps, 2-6
  - supported paths, 2-1
  - upgrading JD Edwards source systems, 8-1
  - upgrading older releases of Oracle BI for Oracle EBS, 6-1
  - upgrading Oracle BI Applications 7.9.x for Siebel Sources, 5-1
  - upgrading PeopleSoft source systems, 7-1
  - upgrading Siebel Analytics 7.5.x, 3-1
  - upgrading Siebel Business Analytics 7.7, 7.7.x, 7.8.x, 4-1
- upgrade considerations
  - points to consider, 2-3

## V

---

- Vehicle Sales
  - upgrading Oracle BI repository, F-1

---