

Oracle® Business Intelligence

New Features Guide

Release 10.1.3.4.2

E10416-07

July 2011

Copyright © 2008, 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 of The Open Group.

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	xiii
Audience	xiii
Documentation Accessibility	xiii
Related Documents	xiii
Conventions	xiv
 Part I General New Features	
 1 Oracle BI Answers-Based Metadata Dictionary Feature in Release 10.1.3.3	
1.1 Overview of Accessing the Metadata Dictionary from Oracle BI Answers	1-1
1.2 Configuration for the Oracle BI Answers-Based Metadata Dictionary Feature	1-1
 2 Multi-Select Dashboard Prompts Feature in Release 10.1.3.3	
2.1 Overview of Multi-Select Dashboard Prompts	2-1
2.2 Configuration for the Multi-Select Dashboard Prompts Feature	2-2
 3 Office Integration with BI Dashboard Feature in Release 10.1.3.3	
3.1 Overview of Integration with Dashboards	3-1
3.2 Configuration for the Office Integration with BI Dashboard Feature	3-1
 4 Dashboard Link Types Feature in Release 10.1.3.3	
4.1 Using Bookmark Links	4-1
4.2 Using Prompted Links	4-2
4.3 Configuration for the Dashboard Prompt Types Feature	4-2
 5 Import Oracle Catalog Feature in Release 10.1.3.3	
 6 Embedded Database Functions Feature in Release 10.1.3.3	
6.1 EVALUATE Function	6-1
6.2 EVALUATE_AGGR Function	6-1
6.3 EVALUATE_PREDICATE Function	6-2

7 Drag and Drop XMLA Metadata Objects Feature in Release 10.1.3.3

8 Catalog Manager Enhancements in Release 10.1.3.3

8.1	General Enhancements for Catalog Manager	8-1
8.2	Copying and Pasting in Catalog Manager	8-2
8.3	General Fixes for Catalog Manager	8-3
8.4	Upgrading Catalogs for EBA	8-3
8.5	Guidelines for Object Names	8-5
8.6	File System Guidelines for Catalogs	8-6
8.6.1	Handling Users of the Catalog.....	8-6
8.6.2	Handling Heterogeneous Nodes.....	8-6
8.6.3	Handling Catalog Files on Various Platforms.....	8-7
8.6.4	Known Issues with Catalog Files	8-7
8.7	Limitations and Workarounds in Catalog Manager.....	8-8

9 General New Features in Release 10.1.3.3.1

9.1	Oracle Business Intelligence Installer Deploys BI Components into One OC4J Container	9-1
9.1.1	Upgrading to 10.1.3.3.1 from a BIEE 10.1.3.x Release Before 10.1.3.3.1	9-1
9.2	DataDirect Connect ODBC Drivers for Linux and UNIX Operating Systems Bundled with Release 10.1.3.3.1	9-2
9.2.1	Using the DataDirect Connect ODBC Drivers	9-2
9.2.1.1	For Connectivity to Microsoft SQL Server Database.....	9-2
9.2.1.2	For Connectivity to Sybase ASE Database.....	9-3
9.2.1.3	For Connectivity to Informix Database	9-4
9.3	Support for Charts in Arabic, Thai, and Hebrew	9-5
9.3.1	Prerequisites	9-5
9.3.2	Integrate the Data Sources.....	9-5
9.3.2.1	Adding the Oracle BI Server as a Data Source	9-5
9.3.2.2	Integrating with Presentation Services.....	9-6
9.3.3	Create a Report and Layout in BI Publisher	9-6
9.3.4	Embed the BI Publisher Report in a Dashboard	9-7
9.3.5	Supported Graph Types	9-8
9.3.6	Known Limitations.....	9-8

10 General New Features in Release 10.1.3.3.2

10.1	Support for Hyperion Essbase as a Data Source	10-1
10.1.1	Supported Versions of Hyperion Essbase with Oracle BI EE 10.1.3.3.2	10-1
10.1.2	Enabling the Use of Hyperion Essbase as a Data Source.....	10-1
10.1.2.1	Essbase Client Libraries.....	10-1
10.1.2.2	BI Server Configuration for UNIX and Linux Platforms.....	10-1
10.1.2.2.1	Solaris: Oracle BI 64-Bit Mode.....	10-1
10.1.2.2.2	Solaris: Oracle BI 32-Bit Mode	10-2
10.1.2.2.3	HP-UX PARISC: Oracle BI 32-Bit Mode.....	10-2
10.1.2.2.4	HP-UX Itanium: Oracle BI 64-Bit Mode	10-2
10.1.2.2.5	AIX Oracle BI 32-Bit and 64-Bit Modes	10-3

10.1.2.2.6	Linux: Oracle BI 32-Bit Mode	10-3
10.1.2.3	Import Process	10-3
10.1.3	Guidelines and Limitations	10-5
10.1.3.1	Measures	10-5
10.1.3.2	Unique Member Names	10-5
10.1.3.3	Time Series Functions	10-6
10.1.3.4	Evaluate Function for MDX Sources.....	10-6
10.1.3.5	User-Defined Attributes	10-7
10.1.3.6	Substitution Variables.....	10-7
10.1.3.7	Attribute Dimensions.....	10-7
10.1.3.8	Unbalanced Hierarchies	10-8
10.2	Using BI EE OC4J Components with IIS Through the IIS Plug-In	10-9
10.2.1	Prerequisites	10-9
10.2.2	Configuration Steps.....	10-10

11 General New Features in Release 10.1.3.3

12 General New Features in Release 10.1.3.4

12.1	Overview of New Features	12-1
12.2	Migration of Oracle BI Discoverer Metadata	12-2
12.3	New Line Chart for Time Series Data	12-2
12.4	New Sample Oracle BI Application and Data Set.....	12-2
12.4.1	OBI EE Web Dashboard Samples.....	12-3
12.4.2	Oracle BI EE RPD Constructs Samples.....	12-4
12.5	Integration with Oracle Enterprise Performance Management Workspace, Fusion Edition	12-5
12.5.1	Overview of the Integration.....	12-5
12.5.2	Documentation for the Integration	12-5
12.5.2.1	Installation Documentation for the Integration	12-6
12.5.2.2	Integration of Oracle BI Answers, Interactive Dashboards, and Delivers, with EPM Workspace	12-6
12.5.2.3	Integration of Oracle BI Publisher with EPM Workspace	12-6
12.5.3	Installing and Configuring Oracle BI Presentation Services with EPM Workspace	12-6
12.5.3.1	Registering the Oracle BI Presentation Services Components with the Hyperion Registry	12-7
12.5.3.2	Specifying the External Login for Single Sign-On with EPM Workspace.....	12-8
12.5.3.3	Configuring Javahost to Enable the Hyperion Common Security System Authenticator	12-9
12.5.3.4	Specifying the Hyperion_Home Environment Variable	12-9
12.5.3.5	Configuring the Repository to Use a Custom Authenticator.....	12-10
12.5.3.6	Configuring the Web Server for EPM Workspace.....	12-11
12.5.3.6.1	Configuring Graphically	12-11
12.5.3.6.2	Configuring Manually	12-12
12.5.3.7	Verifying the Web Server Configuration	12-12
12.5.3.8	Provisioning Hyperion Shared Services Users in Oracle BI Presentation Services	12-12

12.5.4	Using Oracle BI Presentation Services Components in EPM Workspace, Fusion Edition	12-13
12.5.4.1	Specifying Oracle BI Preferences Within EPM Workspace	12-13
12.5.4.2	Performing Oracle BI Presentation Services Administration Tasks Within EPM Workspace	12-13
12.5.4.3	Launching Oracle BI Presentation Services Components in EPM Workspace	12-14
12.5.4.4	Using BI Answers	12-14
12.5.4.5	Using BI Interactive Dashboards.....	12-14
12.5.4.6	Using BI Delivers	12-14
12.6	Integration with Oracle Smart Space, Fusion Edition	12-15
12.6.1	Overview of the Integration	12-15
12.6.2	Documentation for the Integration	12-15
12.6.2.1	Installation Documentation for the Integration	12-15
12.6.2.2	Integration of Oracle BI Answers, Interactive Dashboards, and Delivers with Oracle Smart Space	12-16
12.6.2.3	Integration of Oracle BI Publisher with Oracle Smart Space	12-16
12.6.3	Performing Oracle BI Tasks in Smart Space	12-16
12.6.3.1	Tasks from Right-Click Menus	12-17
12.6.3.2	Folder Management	12-18
12.6.3.3	Oracle BI Delivers iBots	12-18
12.6.3.4	Authentication	12-18
12.6.4	Downloading Oracle Smart Space and Smart View from Oracle BI Presentation Services	12-18
12.6.4.1	Setting Up to Download Oracle Smart Space and Smart View from Oracle BI Presentation Services	12-18
12.6.4.1.1	Smart View	12-18
12.6.4.1.2	Oracle Smart Space.....	12-19
12.6.4.2	Initiating the Download	12-20
12.7	Certification of Oracle WebLogic Server 10.x	12-20
12.7.1	Configuring Oracle BI EE for WebLogic Server.....	12-20
12.7.2	Configuring Oracle BI Publisher for WebLogic Server	12-21

13 General New Features in Release 10.1.3.4.1

13.1	Configuring for Connections to EPM Workspace.....	13-1
13.2	Validating the Catalog.....	13-1
13.2.1	Process: Validating the Catalog	13-1
13.2.2	Important Guidelines for Validating the Catalog.....	13-2
13.2.3	Performing a Basic Validation of the Catalog.....	13-2
13.2.4	Specifying the Elements for Validating the Catalog.....	13-3

14 General New Features in Release 10.1.3.4.2

14.1	Support for SAP/BW Variables	14-1
14.2	Enhancements for Catalog Manager	14-2
14.2.1	General Enhancements	14-2
14.2.2	Creating Reports	14-2
14.2.3	Obtaining Audit Trail Information for Objects	14-3

Part II New Features for Oracle BI Publisher

15 Oracle BI Publisher Template Builder Enhancements in Release 10.1.3.3

15.1	Changes to the Oracle BI Publisher Menu	15-1
15.2	Integration with Oracle BI Discoverer	15-2
15.3	Microsoft PowerPoint Output Support	15-2
15.4	Advanced Form Field Support	15-3
15.4.1	Edit Data Fields	15-3
15.4.2	Advanced Tab	15-4
15.4.3	Insert and Edit Repeating Groups	15-5
15.4.3.1	Creating Repeating Groups	15-5
15.4.3.2	Create Grouping Fields Around a Block	15-6
15.4.4	Insert Conditional Formatting	15-7
15.4.5	Inserting and Editing Conditional Regions	15-8
15.5	Chart Builder Enhancements	15-9
15.5.1	Support for Charts Inside a Grouping	15-9
15.5.2	New Chart Types	15-10
15.5.3	New Chart Subtypes	15-11

16 Flash Templates for Oracle BI Publisher in Release 10.1.3.3

16.1	Changes to the BI Publisher User Interface	16-1
16.2	Prerequisites	16-2
16.3	Building Templates	16-2
16.3.1	Adding the Data Source	16-2
16.3.2	Creating the Layout	16-5
16.3.2.1	Adding a Chart	16-6
16.3.3	Data Binding	16-6
16.3.3.1	Binding the DataGrid	16-7
16.3.3.2	Binding the Chart	16-8
16.4	Loading Templates	16-9
16.5	Setting Properties for PDF Output	16-9
16.6	Summary	16-10

17 Oracle BI Publisher PowerPoint Output in Release 10.1.3.3

17.1	Supported Versions of PowerPoint	17-1
17.2	Configuring Fonts for BI Publisher Server	17-1
17.3	Configuring Fonts for BI Publisher Template Builder	17-2
17.4	Usage and Limitations	17-2

18 Oracle BI Publisher Integration with Oracle BI Discoverer in Release 10.1.3.3

18.1	Configuring the Oracle BI Publisher and Discoverer Integration	18-1
18.1.1	Step 1: Install the Discoverer Web Service	18-2
18.1.2	Step 2: Install Oracle BI Publisher 10.1.3	18-2

18.1.3	Step 3: Configure Discoverer to Use Oracle Application Server Identity Management	18-2
18.1.4	Step 4: Set Up a Dedicated User to Access the Discoverer Web Service from Oracle BI Publisher	18-2
18.1.5	Step 5: Configure Oracle BI Publisher to Use the Discoverer Web Service.....	18-3
18.1.6	Discoverer Web Service Configuration Settings	18-4
18.2	Creating Reports Based on Discoverer Worksheets	18-4
18.2.1	About Creating a Report Based on a Discoverer Worksheet	18-4
18.2.2	Creating a Report Directly Through the Oracle BI Publisher Web Interface.....	18-5
18.2.3	Creating a Report Indirectly Through Oracle BI Publisher Template Builder	18-8
18.2.3.1	About Creating Layout Templates.....	18-11
18.2.3.2	Creating a Layout Template Offline	18-11
18.2.3.3	Creating a Layout Template Online	18-18
18.2.3.4	Viewing and Scheduling Reports.....	18-20
18.2.4	Prerequisites, Limitations, and Known Issues	18-20
18.2.4.1	Prerequisites	18-20
18.2.4.2	Limitations.....	18-21
18.2.4.3	Known Issues	18-21

19 Performance Improvements for Oracle BI Publisher in Release 10.1.3.3

19.1	Multithreading for Bursting	19-1
19.2	Multithreading for Document Generation.....	19-2

20 Converting Reports from Oracle Reports to Oracle BI Publisher in Release 10.1.3.3

20.1	Overview	20-1
20.2	Prerequisites	20-2
20.2.1	Reports and Oracle Reports XML Format.....	20-2
20.2.2	Update CLASSPATH with Needed JAR Files.....	20-2
20.2.3	JDK Version 1.1.8	20-2
20.3	Conversion Utility.....	20-2
20.3.1	Output Files	20-3
20.4	Upload PL/SQL Package to the Database	20-4
20.5	Putting Converted Reports into the BI Publisher Repository	20-4
20.6	Testing and Tweaking the Reports.....	20-5
20.6.1	Data Template and PL/SQL Files	20-5
20.6.2	RTF Layout Template File	20-5

21 Support for Digital Signature in PDF Documents in Release 10.1.3.3

21.1	Introduction	21-1
21.2	Prerequisites	21-1
21.3	Limitations	21-2
21.4	Integration with BI Publisher	21-2
21.4.1	Digital Signature ID Registration and Allowed Role Assignment.....	21-2
21.4.2	Specify the Signature Display Field or Location	21-4
21.4.2.1	Designating a Field in a PDF Template for the Digital Signature	21-4

21.4.2.2	Specifying a Location in the Document to Insert the Digital Signature (for RTF and PDF Templates)	21-6
21.4.3	Running and Signing Reports with the Digital Signature	21-7
21.4.4	Additional PDF Security Features	21-8
22	Support for PostScript Printers in Release 10.1.3.3	
22.1	Introduction	22-1
22.2	Setting Up the Print Server to Use the Converter	22-1
22.3	Limitations	22-2
23	Additional New Features in BI Publisher Release 10.1.3.3	
23.1	Support for CSV Output	23-1
23.2	Report Definition Features	23-1
23.3	Report Viewing Features	23-2
23.4	New Property for LDAP Security Model	23-2
24	New Features in BI Publisher Release 10.1.3.3.1	
24.1	Time Out Value for Web Service Data Sources	24-1
24.2	Bidirectional Function for eText Data Fields	24-1
24.3	Default Repository	24-2
25	New Features in BI Publisher Release 10.1.3.3.2	
25.1	Support for Document Cache.....	25-1
25.2	Support for New Output Types.....	25-1
26	New Features in BI Publisher Release 10.1.3.3.3	
26.1	BI Publisher Scheduler Setup Updates Including the Bundling of DataDirect Connect for JDBC Drivers	26-1
26.1.1	Recommendations for Using DataDirect Connect or Native Database Drivers.....	26-1
26.1.2	Setting Up a Connection to Your Scheduler Database Using the DataDirect Connect for JDBC Drivers	26-2
26.1.2.1	Entering Connection Information for IBM DB2	26-2
26.1.2.2	Entering Connection Information for Microsoft SQL Server	26-3
26.1.2.3	Entering Connection Information for Oracle	26-3
26.1.2.4	Entering Connection Information for MySQL	26-3
26.1.3	Setting the BLOB Column Size	26-4
26.2	"Link to This Report" Can Now Be Set as a Report Option	26-4
26.3	Enhanced Support for Pagination Control in RTF Output	26-4
26.3.1	Support for "Keep Paragraph with the Next Paragraph"	26-5
26.3.2	Keep Paragraph Intact.....	26-5
26.3.3	Support for Table "Heading Rows Repeat"	26-5
26.3.4	Support for "Table Row Keep Together"	26-5
26.4	Support for Multiple Cascading Parameters	26-5
26.5	Support for Vertical Alignment of External Graphics.....	26-6
26.6	Support for Multiple Currency Symbol Displays in Report.....	26-6

26.6.1	Sample Usage	26-7
--------	--------------------	------

27 BI Publisher Web Services, Releases 10.1.3.3.1, 10.1.3.3.2, and 10.1.3.3.3

27.1	Overview of the Oracle BI Publisher Web Services	27-1
27.1.1	Functionality Overview	27-1
27.1.2	WSDL Definition for PublicReportService.....	27-1
27.2	Oracle BI Publisher Web Services Data Types.....	27-2
27.2.1	Base Data Types	27-2
27.2.2	XML-to-Java Data Type Mappings	27-2
27.2.3	Complex Types	27-2
27.3	Oracle BI Publisher Web Services Operations	27-7
27.3.1	About Using the Code Samples.....	27-7
27.3.2	Operations for Validation of Privileges.....	27-8
27.3.3	Operations to Get Information About Reports and the Repository	27-10
27.3.4	Operations to Run and Schedule Reports	27-13
27.3.5	Messages for Errors	27-17
27.4	Debugging Web Service Applications	27-17
27.5	Oracle BI Publisher Web Services WSDL	27-19

28 New Features in BI Publisher Release 10.1.3.4

28.1	Integration with Oracle Enterprise Performance Management Workspace	28-1
28.2	Integration with Oracle Smart Space	28-1
28.3	Support for Multidimensional Data Sources	28-2
28.4	Enhancements to the BI Publisher Web Service	28-2
28.5	Support for "After Report" Triggers.....	28-3
28.6	Support for Complex Web Services as a Data Source	28-3
28.7	New Report Property "Disable Client Access from Analyzer for Excel"	28-3
28.8	Enhancements to the Template Builder for Microsoft Word Add-in.....	28-4

29 New Features in BI Publisher Release 10.1.3.4.1

30 New Features in BI Publisher Release 10.1.3.4.2

Part III New Features for Oracle Business Intelligence for Microsoft Office

31 Installing Oracle BI for Microsoft Office in Release 10.1.3.3

31.1	System Requirements	31-1
31.1.1	Supported Client Operating Systems	31-1
31.1.2	Supported Versions of Microsoft Office	31-1
31.1.3	Supported Application Servers	31-2
31.2	Installing the BI Office Server	31-2
31.2.1	Deployment of the bioffice.ear.....	31-2
31.2.2	Initialization of the Server Configuration File.....	31-2
31.2.3	Location of the Installed Client OracleBIOffice.exe File	31-3
31.3	Deploying the BI Office Server on J2EE Application Servers.....	31-3
31.4	Installing the Oracle BI Add-In for Microsoft Office	31-4

31.4.1	Performing a Maintenance or Upgrade Install.....	31-5
31.4.2	Configuring the Client	31-6
31.5	Setting Properties in the biooffice.xml Configuration File.....	31-7
31.6	Setting Up SSL-Enabled OC4J for Oracle BI for Microsoft Office.....	31-11
31.7	Setting Up an SSO-Enabled Presentation Server for Releases Before 10.1.3.4	31-14
31.8	Setting Up an SSO-Enabled Presentation Server for Releases 10.1.3.4 and Later.....	31-15
31.8.1	Prerequisites	31-15
31.8.2	Integrating Oracle BI for Microsoft Office with SSO-Enabled Oracle BI EE	31-16
31.8.3	Alternative Configuration Method	31-17
31.9	Setting Up an NTLM-Enabled Presentation Server for Releases 10.1.3.4 and Later....	31-18
31.9.1	Example of Deploying and Configuring with NTLM Authentication	31-19
31.9.2	Prerequisite Tasks.....	31-20
31.9.2.1	Configure Oracle BI EE to Enable Single Sign-On.....	31-20
31.9.2.2	Optionally Install and Configure Active Directory or Another Third-Party LDAP Server	31-20
31.9.2.3	Create a Second BI Presentation Services Servlet to Bypass Authentication	31-20
31.9.2.3.1	Method 1: Install a Standalone Instance of the Web Server and Deploy and Configure the Presentation Services Plug-in	31-20
31.9.2.3.2	Method 2: Create a Duplicate SAWBridge Servlet in the web.XML File	31-21
31.9.3	Installing the NTLM Module	31-21
31.9.4	Configuring Oracle BI for Microsoft Office to Enable NTML.....	31-23
31.9.5	Testing the NTLM Deployment.....	31-24

32 Microsoft Office Support in Release 10.1.3.3

33 New Features in Oracle BI for Microsoft Office Release 10.1.3.3.1

33.1	Configuring the Connection URL.....	33-1
33.2	Language Support.....	33-1

34 New Features in Oracle BI for Microsoft Office Release 10.1.3.3.3

35 New Features in Oracle BI for Microsoft Office Release 10.1.3.4

35.1	Server Independence	35-1
35.2	Changes to the Add-In	35-1
35.2.1	Ability to Enable Logging.....	35-1
35.2.2	Enhanced BI Catalog Task Pane	35-2
35.2.3	Ability to Preserve Excel Conditional Formatting During Refreshes	35-2
35.2.4	Support for Gauge View	35-2
35.2.5	Support for Funnel View	35-2
35.2.6	Edit Prompts and Levels After Insert	35-2
35.2.7	Enhanced Prompts Selection User Interface	35-2
35.2.8	Support for Grand Totals in Table Views	35-3
35.3	Support for Combined Request Views	35-3
35.4	Transparent Compression of Results	35-3

35.5	NTLM Support	35-3
36	New Features in Oracle BI for Microsoft Office Release 10.1.3.4.1	
37	New Features in Oracle BI for Microsoft Office Release 10.1.3.4.2	

Preface

This guide describes the new features in Releases 10.1.3.3, 10.1.3.3.1, 10.1.3.3.2, 10.1.3.3.3, 10.1.3.4, 10.1.3.4.1, and 10.1.3.4.2 of Oracle Business Intelligence Enterprise Edition (Oracle BI Enterprise Edition), Oracle Business Intelligence Publisher (Oracle BI Publisher), and Oracle Business Intelligence for Microsoft Office (Oracle BI for Microsoft Office).

Audience

This document is intended for:

- Oracle BI Enterprise Edition report and dashboard creators
- Oracle BI Enterprise Edition administrators
- Oracle Business Intelligence Publisher administrators, report authors, and end users

Documentation Accessibility

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

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

More documentation about Oracle BI Enterprise Edition is available as follows:

- Within the product, as online help.
- On OTN, at the Oracle BI Enterprise Edition Documentation page:
http://www.oracle.com/technology/documentation/bi_ee.html
- On OTN, at the Oracle BI Publisher Documentation page:
http://www.oracle.com/technology/documentation/bi_pub.html

For detailed information about new features and functionality in Oracle Business Intelligence Publisher Release 10.1.3.4.1 and later, refer to *Oracle Business Intelligence Publisher New Features Guide*.

You can also find additional information at the following locations

- The Oracle BI Enterprise Edition Product Web site:
<http://www.oracle.com/technology/products/bi/enterprise-edition.html>
- The Oracle Business Intelligence Publisher Product Web site:
<http://www.oracle.com/technology/products/xml-publisher/index.html>

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.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Part I

General New Features

Part I contains the following chapters:

- [Chapter 1, "Oracle BI Answers-Based Metadata Dictionary Feature in Release 10.1.3.3"](#)
- [Chapter 2, "Multi-Select Dashboard Prompts Feature in Release 10.1.3.3"](#)
- [Chapter 3, "Office Integration with BI Dashboard Feature in Release 10.1.3.3"](#)
- [Chapter 4, "Dashboard Link Types Feature in Release 10.1.3.3"](#)
- [Chapter 5, "Import Oracle Catalog Feature in Release 10.1.3.3"](#)
- [Chapter 6, "Embedded Database Functions Feature in Release 10.1.3.3"](#)
- [Chapter 7, "Drag and Drop XMLA Metadata Objects Feature in Release 10.1.3.3"](#)
- [Chapter 8, "Catalog Manager Enhancements in Release 10.1.3.3"](#)
- [Chapter 9, "General New Features in Release 10.1.3.3.1"](#)
- [Chapter 10, "General New Features in Release 10.1.3.3.2"](#)
- [Chapter 11, "General New Features in Release 10.1.3.3.3"](#)
- [Chapter 12, "General New Features in Release 10.1.3.4"](#)
- [Chapter 13, "General New Features in Release 10.1.3.4.1"](#)
- [Chapter 14, "General New Features in Release 10.1.3.4.2"](#)

Oracle BI Answers-Based Metadata Dictionary Feature in Release 10.1.3.3

This new feature enables administrators to analyze metadata repository statistics in Oracle BI Answers. This chapter contains the following sections:

- [Section 1.1, "Overview of Accessing the Metadata Dictionary from Oracle BI Answers"](#)
- [Section 1.2, "Configuration for the Oracle BI Answers-Based Metadata Dictionary Feature"](#)

1.1 Overview of Accessing the Metadata Dictionary from Oracle BI Answers

The Oracle BI Server can generate a Metadata Dictionary that describes the metrics contained within the repository and the attributes of repository objects. The Metadata Dictionary output is a static set of XML documents. For more information about the Metadata Dictionary, see *Oracle Business Intelligence Server Administration Guide*.

In this release, this Metadata Dictionary can be accessed directly from the Oracle BI Answers selection pane where specific metadata information is shown to guide report construction. Once this Oracle BI Answers-based functionality has been enabled as described in this section, an icon is displayed next to each Subject Area, Table name, and Column shown in the selection pane for that Subject Area. The Subject Area icon is always displayed, whereas the Table name and Column name icons are only displayed on mouse-over. Clicking on the icon displays a specific Metadata Dictionary page to show information for that element and links to related repository information.

1.2 Configuration for the Oracle BI Answers-Based Metadata Dictionary Feature

This section explains how to configure the Oracle BI Answers Based Metadata Dictionary feature.

To configure this functionality, a Metadata Dictionary should first be generated using the Oracle BI Administration Tool, as follows:

1. On the computer where Oracle BI is installed, launch the Oracle BI Administration Tool.
2. Log in to the desired repository in "Offline" mode.
3. Create the dictionary using: **Tools -> Utilities -> Generate Metadata Dictionary**.

Once the newly-generated Metadata Dictionary has been saved, it should then be packaged and deployed to the application server as appropriate for your installation. If you are using IIS as a web server, then placing the Metadata Dictionary in the "OracleBIData" directory is a recommended practice (for example, within OracleBIData\web\res\dictionary). Any http-based location also works.

The following "SubjectAreaMetadata" node must be added to the Presentation Services instanceconfig.xml file to specify this selected location. Add the element under the "ServerInstance" node. See the "Making Oracle BI Presentation Services Configuration Changes" section within *Oracle Business Intelligence Presentation Services Administration Guide* for more information on editing the instanceconfig.xml configuration file.

Here is an example showing a relative reference to the IIS location previously mentioned. For any other application server this path would point to where the WAR file had been deployed.

```
<SubjectAreaMetadata>
<DictionaryURLPrefix>/analyticsres/dictionary/</DictionaryURLPrefix>
</SubjectAreaMetadata>
```

Generate the Metadata Dictionary, make this configuration change, then restart Presentation Services to use this feature. Best practices include keeping repository names to less than 156 characters to prevent truncated URLs. Access to the Metadata Dictionary through Oracle BI Answers can also be administered using a new Access to Metadata Dictionary privilege. Finally, some versions of browsers (such as Internet Explorer 6 or Firefox 1.6) might have problems showing these XML-based files if they are not configured to access data sources across domains. This setting should be enabled.

Multi-Select Dashboard Prompts Feature in Release 10.1.3.3

This new feature enables users to analyze data more easily with an easier to use and more powerful screen for specifying dashboard prompts. For example, users can now use multi-select to specify prompt values, and a more powerful search facility.

This chapter contains the following sections:

- [Section 2.1, "Overview of Multi-Select Dashboard Prompts"](#)
- [Section 2.2, "Configuration for the Multi-Select Dashboard Prompts Feature"](#)

2.1 Overview of Multi-Select Dashboard Prompts

Prior to this release, a multi-select dashboard prompt with a large number of values could become difficult to work with. A new search feature adds the ability to better navigate and select values from a multi-select prompt when there are many (possibly hundreds or thousands) of values. Modifications to the existing multi-select dashboard prompt now allow the user to search through the available values using four matching schemes: begins with, ends with, contains, and is Like (Pattern Match). Wildcards such as the percent sign (%) can also be used in the query. A more sophisticated paging mechanism is also now present that allows the user to page ahead according to a configured search set size as needed. A plus sign following the "Choices Returned" number and the "More..." button indicate additional values have not yet been shown. Configurations for the search set size are described in the following paragraphs.

As with the previous multi-select prompt, to add values to be filtered, choices are selected and moved using the shuttle buttons to the left pane. A single mouse click selects a single value. Multiple values can be selected by clicking while holding down the Ctrl key. Range selection is done by holding down the Shift key while clicking. Buttons then allow the user to move single values, these sets of values, or all visible values to or from the "Selected" pane.

Once in the "Selected" pane, the collection of values can be edited by clicking on the Edit button. This action displays an editable dialog where new values can be typed in or lists of values can be copied and pasted in or out of the prompt. A list of values in Excel, for example, can easily be pasted in and thus "imported" into the prompt. This edit dialog interprets sequences of values separated by carriage returns or line feeds, tab delimiters, and pipe (|) characters. Note that while manually typing in values, only carriage returns/line feeds and "|" can be used as value separators. Tab is a valid value separator only when the user copies and pastes values that are tab-separated.

2.2 Configuration for the Multi-Select Dashboard Prompts Feature

To configure this feature, edit the `instanceconfig.xml` file to specify the number of values returned per search set. Add the `<Prompts>` and `<MaxScrollValues>` elements within the `<ServerInstance>` node, as shown in the following example:

```
<Prompts>  
<MaxScrollValues>20</MaxScrollValues>  
</Prompts>
```

See the "Making Oracle BI Presentation Services Configuration Changes" section in *Oracle Business Intelligence Presentation Services Administration Guide* for complete information on editing the `instanceconfig.xml` configuration file.

Office Integration with BI Dashboard Feature in Release 10.1.3.3

This new feature enables users to save reports in PowerPoint format and to paste reports directly into Microsoft Office applications.

This chapter contains the following sections:

- [Section 3.1, "Overview of Integration with Dashboards"](#)
- [Section 3.2, "Configuration for the Office Integration with BI Dashboard Feature"](#)

3.1 Overview of Integration with Dashboards

Two new Interactive Dashboard features have been added to this release that provide additional interactivity with Microsoft Office. These features complement the new Oracle BI Add-ins for PowerPoint and Excel that provide functionality natively within those Office applications. For more information on the Oracle BI Add-ins for PowerPoint and Excel, see [Chapter 31, "Installing Oracle BI for Microsoft Office in Release 10.1.3.3"](#) and [Chapter 32, "Microsoft Office Support in Release 10.1.3.3"](#).

The first new Dashboard feature is "Download to PowerPoint", which adds PowerPoint alongside Excel as a targeted application for downloaded reports. This new link is found within the "Report Links" Download menu and downloads a static version of that report to PowerPoint. Like the other Download options, when the link is first clicked, the user has the choice of directly saving the report to disk or opening it immediately. The add-in need not be installed to use this functionality.

The second new Dashboard feature provides a means to copy and paste Dashboard reports directly into Office applications. This functionality is available along with the other Dashboard Report Links configured using the Dashboard Editor. A new "Copy" link can be added to a report that copies the XML definition of the current report to the Windows clipboard. Pasting this link using the Paste function in the Oracle BI menu converts the copied XML into an Office version of that report. Naturally this functionality is present only in versions of either PowerPoint or Excel in which the Oracle BI Add-in has been installed. Once pasted, this Dashboard report is translated into native Office objects that can be subsequently refreshed with live data at any point. Pasting the link directly using Edit/Paste either in Office or any other application (such as a text editor) copies the request XML without translation.

3.2 Configuration for the Office Integration with BI Dashboard Feature

Access to both of these Dashboard Report Links can be administered using the Access to Oracle BI Office Add-in privilege.

Dashboard Link Types Feature in Release 10.1.3.3

This new feature enables users to share reports and collaborate more easily by providing shortcut links to them, including prompts.

Two new types of Dashboard links can be generated using this release. Intended for collaboration, these are the:

- Bookmark Link
- Prompted Link

This chapter contains the following sections:

- [Section 4.1, "Using Bookmark Links"](#)
- [Section 4.2, "Using Prompted Links"](#)
- [Section 4.3, "Configuration for the Dashboard Prompt Types Feature"](#)

4.1 Using Bookmark Links

The Bookmark Link provides a mechanism whereby a user can obtain a shortcut link to a Dashboard page, which contains all aspects of page state. This link can then be e-mailed to another user who (assuming they had identical permissions and access) can enter that link into the URL line of a browser and view the same page content. The recipient can then further analyze the data in the Dashboard. Technically, page state is too complex to capture in a URL so this functionality writes an object to the Presentation Services Catalog, which is called by the Bookmark Link URL. To create a Bookmark Link the user selects the "Create Bookmark Link" from the Dashboard "Page Options" menu. Once selected, the Dashboard message bar shows that "A Bookmark Link suitable for saving or sharing has been created for this page. This Link can be found in the browser's Address Bar." Inspection reveals that the URL has indeed changed in the Address Bar to show the Bookmark Link. At this point the user can copy and paste this link from the Address Bar and either save it or share it most likely through e-mail. After a Bookmark Link has been generated, the user can continue to modify the content of the page, such as adding or deleting reports. The Bookmark link tries to match context between the Bookmark Link and the Dashboard Page. The existing Dashboard Saved Selections functionality exploits a similar matching mechanism. For example, if a specific filter on the original Dashboard page were included in the saved Bookmark Link, but all reports that contain that filter were later removed from that Dashboard, then the Bookmark Link still works. The Bookmark Link simply contains a small amount of extraneous information, which is not applied. All other page context still appropriate is applied.

4.2 Using Prompted Links

A second variant of this functionality generates a Prompted Link, which does not capture all aspects of state, but instead provides a simplified link to the Dashboard Page, which contains both the path to that page and a simplified presentation of Dashboard Prompts. This Prompted Link is intended to support both manual and programmatic manipulation of that link and aims to provide a simple URL-based API for generating and customizing Dashboard content. Unlike the Bookmark Link, this link is dynamic, which maps to no saved catalog content.

When configured to show this feature, the Dashboard Page Options menu includes a "Create Prompted Link" option. Selecting this option generates the Prompted Link and the user sees a message that indicates, "A Prompted link capturing the prompts and values of this page has been created."

This is the Prompted Link syntax:

```
<Content Path>, Action=Navigate, Column1 (col1), Operator1 (op1), Values1(val1),
Column2 (col2), Operator2 (op2), Values2 (val2) ...
```

Values are separated by plus signs (+) and to further simplify this syntax the operator parameter is not needed for equal (eq). Double quotes are optional if there are no white spaces inside the values.

The following are Prompted Link examples:

```
...Action=Navigate&col1=Products.Color&val1=Armory
...Action=Navigate&col1=Products.Color&val1=Armory+Clear
...Action=Navigate&col1=Products.Color&op1=bet&val1=Armory+Clear
...Action=Navigate&col1=Products.Color&val1="Armory"+"Clear"&col2=Periods."Month"&
val2="04/01/1998%2012:00:00%20AM"
...Action=Navigate&col1=Products.Color&val1="Armory"+"Clear"&col2=Periods."Month"&
op2=gt&val2="05/01/1998%2012:00:00%20AM"
```

Here is an example of the Full URL:

```
@http://localhost/aruba/saw.dll?Dashboard&PortalPath=%2Fshared%2Fnew%2F_
portal%2Fprompt&Page=page%252&Action=Navigate&col1=Products.Color&val1="Armory"+"C
lear"&col2=Periods."Month"&op2=gt&val2="05/01/1998%2012:00:00%20AM"
```

As these links are extensions of the existing Go URL, a full list of operators can be found in the "Using the Oracle BI Presentation Services Go URL to Issue SQL and Pass Filters" section found in *Oracle Business Intelligence Presentation Services Administration Guide*.

4.3 Configuration for the Dashboard Prompt Types Feature

This section explains how to configure the new Dashboard Prompt Types feature.

For configuration, the Create Bookmark Link is enabled by default. The following nodes must be added to the instanceconfig.xml file under the "ServerInstance" node:

```
<Dashboard>
<EnableBookmarkURL>True</EnableBookmarkURL>
<EnablePromptedURL>True</EnablePromptedURL>
<BookmarkExpirationDays>30</BookmarkExpirationDays>
</Dashboard>
```

"EnableBookmarkURL" controls whether the "Create Bookmark Link" is shown in the "Page Options" menu while "EnablePromptedURL" controls the appearance of the "Create Prompted Link." The Bookmark Link refers to a stub saved in the Presentation

Services Catalog and "BookmarkExpirationDays" allows an administrator to set a lifespan for how long these should persist.

See the "Making Oracle BI Presentation Services Configuration Changes" section within the *Oracle Business Intelligence Presentation Services Administration Guide* for more information on editing the instanceconfig.xml file.

For advanced applications, and for both the "Create Bookmark Link" and "Create Prompted Link" functionality, two Javascript functions can be called to retrieve these values from custom scripts embedded on the originating Dashboard Page. Note, for Javascript functions to work, a HardenXSSconfiguration must be set to false in the instanceconfig.xml file.

For the Bookmark Link, the function linkToPage(bInlineDrill,bGetBookmarkOnly) can be called. If the second argument is omitted or set to "false," then the browser address bar is updated with the bookmark URL. If the second argument is set to "true," then the browser address bar is not refreshed. In either case, the bookmark URL is retrieved asynchronously and put into the variable saw.bookmarkURL. Polling the value of saw.bookmarkURL gives the link. This variable is set to an empty string when the function is invoked. As for the first argument, if you see the "Page Options" menu on the page, then set it to true, otherwise, set it to false.

For the Prompted Link, the function GetPURL() returns the Prompted Link as a string.

Import Oracle Catalog Feature in Release 10.1.3.3

This new feature enables administrators to import Database objects through a native Oracle OCI connection.

The Oracle BI Administration Tool now supports importing of physical Database objects through a native Oracle OCI connection and reuses the same OCI connection for queries. This feature mitigates the need to configure an ODBC connection to an Oracle Database purely for importing metadata objects into the Administration Tool.

Embedded Database Functions Feature in Release 10.1.3.3

This new feature enables users and administrators to create more powerful reports by directly calling Database functions from either Oracle BI Answers or by using a Logical column (in the Logical Table source) within the Metadata (repository).

The Oracle BI server now supports the capability to directly call functions defined within the Database from either the Answers interface or using a Logical column (in the Logical Table source) within the Metadata (repository). This feature is restricted to support SQL sources in this release.

Note: MDX or MOLAP data sources are not supported in this release.

The following new database functions are supported:

- **EVALUATE**
For more information, see [Section 6.1, "EVALUATE Function"](#).
- **EVALUATE_AGGR**
For more information, see [Section 6.2, "EVALUATE_AGGR Function"](#).
- **EVALUATE_PREDICATE**
For more information, see [Section 6.3, "EVALUATE_PREDICATE Function"](#).

6.1 EVALUATE Function

This function is intended for scalar and analytic calculations.

Syntax: `EVALUATE('DB_Function(%1)', <Comma separated Expression>)`

Example: `SELECT e.lastname,sales.revenue,EVALUATE('dense_rank()
over(order by %1)',sales.revenue) FROM sales s, employee e;`

6.2 EVALUATE_AGGR Function

This function is intended for aggregate functions with group by clause.

Syntax: `EVALUATE_AGGR('DB_Aggregate_Function(%1)', <comma
separated Expression>)`

Example: `SELECT year.year, sales.qtysold, EVALUATE_
AGGR('sum(%1)', sales.quantity) From SnowFlakeSales;`

6.3 EVALUATE_PREDICATE Function

This function is intended for functions with a return type of Boolean.

Syntax: `EVALUATE_PREDICATE('DB_Function(%1)', <comma separated
Expression>)`

Example: `SELECT year, Sales as DOUBLE,CAST(EVALUATE('OLAP_
EXPRESSION(%1,'LAG(units_cube_sales, 1, time, time LEVELREL
time_levelrel)'))', OLAP_CALC) AS DOUBLE) FROM "Global".Time,
"Global"."Facts - sales" WHERE EVALUATE_PREDICATE('OLAP_
CONDITION(%1, 'LIMIT time KEEP ''1'', ''2'', ''3'',
''4'' ''') =1', OLAP_CALC) order by year;`

Drag and Drop XMLA Metadata Objects Feature in Release 10.1.3.3

The Oracle BI Administration Tool supports the drag and drop feature for XMLA data sources (SAP/BW and Microsoft Analysis Server). Using this capability, you can drag and drop physical metadata objects (cubes) into the Business Model and Mapping layer as fully configured Logical Model Metadata (Subject area), which retains metrics, attributes, and dimensions. This capability mitigates the need to redefine manually logical dimensional models from data sources, which contain rich dimensional metadata constructs.

Catalog Manager Enhancements in Release 10.1.3.3

Various fixes and enhancements have been added to Catalog Manager, as described in this chapter. This chapter applies to Releases 10.1.3.3, 10.1.3.3.1, 10.1.3.3.2, 10.1.3.3.3, 10.1.3.4, and 10.1.3.4.1 and 10.1.3.4.2.

This chapter contains the following sections:

- [Section 8.1, "General Enhancements for Catalog Manager"](#)
- [Section 8.2, "Copying and Pasting in Catalog Manager"](#)
- [Section 8.3, "General Fixes for Catalog Manager"](#)
- [Section 8.4, "Upgrading Catalogs for EBA"](#)
- [Section 8.5, "Guidelines for Object Names"](#)
- [Section 8.6, "File System Guidelines for Catalogs"](#)
- [Section 8.7, "Limitations and Workarounds in Catalog Manager"](#)

8.1 General Enhancements for Catalog Manager

Oracle BI EE Release 10.1.3.3 includes the following enhancements for Catalog Manager:

- **Popup Menus**

The popup menu on items in the Tree view is now the same as the popup menu in the Table view, with improved context sensitivity.

- **Multiple Selection**

You can now select multiple items and update their properties or permissions simultaneously. If any of the selected items are a folder, then you can also apply those changes recursively to all the items in that folder's tree.

For example, you can set all files under /shared/DontTouch to be Read Only. Select the folder DontTouch, display the popup menu, and select **Properties**. In the dialog, select the **Read Only** box, then select the **Apply recursively** box, and click **OK**.

- **Set Ownership**

If you have permission, then you can set the ownership (using the Properties dialog) of items to a particular user, rather than previously when you could set ownership only to yourself (also known as take ownership).

- Archiving and Unarchiving

Select Main Menu, then File to access the archiving features. The menu is context-sensitive, allowing you to select what to archive and where to unarchive it to. Also see paste options below. Archive now also works on the root folder: "/". Properties, and attributes of items are now also archived.

- Menu options **Smart Rename** and **Search & Replace**

These options have been changed from using regular expressions (to search for a match) to instead use the more intuitive method of the supplied text "as is". For power users, the Search & Replace XML file continues to use regular expressions, so extra care must be taken to ensure that those search strings are correct.

8.2 Copying and Pasting in Catalog Manager

The **Copy to another catalog** and **Paste from another catalog** options are now handled seamlessly by the normal **Copy** and **Paste** options. Similarly, you can now also drag and drop from one Catalog Manager to another. Note that drag and drop always makes a copy of the dragged items, even when performing a drag and drop within a single Catalog Manager. There are two new options for pasting, unarchiving, and dropping. You can set two preferences in the Preferences dialog from the Tools menu *before* pasting.

The Paste Overwrite options in the Preferences dialog are as follows:

- None: Pastes all files, but does not paste any existing files.
- Old: Pastes all files, but does not paste any existing files unless they are older than the source.
- All: Paste all files, whether they exist or not. The default is 'all', which was the default behavior in Release 10.1.3.2.

For example:

Under "/users/joe" are the reports "A" (created 01-Jan-2006), "B" (created 31-May-2006) and "C" (created 01-Jan-2006). Under "/users/sue" are the reports "A" (created 28-Feb-2006) and "B" (created 01-Jan-2006) (and no report "C"). Suppose Sue copies the reports 'A', 'B' and 'C' from /users/joe, and pastes them to /users/sue. If the preference is set to "None", then Sue keeps her 'A' and 'B' reports and Joe's reports are ignored. Sue gets a copy of 'C'. If the preference is set to "All", then Sue's 'A' and 'B' reports are overwritten with Joe's, and Sue gets a copy of 'C'. If the preference is set to "Old", then Sue keeps her 'A' (Sue's 'A' is not "old"), Sue's B gets overwritten by Joe's (Sue's 'B' was "old"), and Sue gets a copy of 'C'.

The Paste ACL options are as follows:

- Inherit pasted item: Inherits its permissions (ACL) from its new parent folder.
- Preserve pasted item: Preserves its permissions (ACL) as it was in the original, mapping accounts as necessary.
- Create pasted item: Preserves its permissions (ACL) as it was in the original, creating and mapping accounts as necessary.

The default is 'Inherit pasted item', which was the default behavior in 10.1.3.2.

Suppose Steve owns "/users/steve/MyFavReport" and has permissions (ACL) "everyone can read/execute, steve has full control". Joe (who has some administration privileges) logs in and copies MyFavReport, pasting it to "/users/sue" (which is owned by "administrator", with permissions "admins have full control, Sue has full

control"). If Joe sets "Inherit", /users/sue/MyFavReport is owned by Joe with whatever permissions are set on "/users/sue" (that is, "administrators have full control, sue has full control").

If Joe sets "Preserve", /users/sue/MyFavReport is owned by Joe with whatever permissions were set on "/users/steve/MyFavReport" (that is, "everyone can read/execute, steve has full control"). If Joe pastes in a second Catalog Manager and if "steve" does not exist in this Catalog, then the permissions for Steve are discarded.

If "steve" exists but has a different user ID, then Steve's user ID is mapped to the new one. If Joe sets "Create", /users/sue/MyFavReport is owned by Joe with whatever permissions were set on "/users/steve/MyFavReport" (that is, "everyone can read/execute, Steve has full control"). If Joe pastes in a second Catalog Manager and if "steve" does not exist in this Catalog, then the user "steve" is created. This is used by OnDemand, where they create accounts in a staging area before moving the user to the production environment.

After the paste, the object is owned by the person doing the paste. Thus you must likely select the newly pasted item and set ownership recursively to whomever you want (if you have permission to do that).

8.3 General Fixes for Catalog Manager

Oracle BI EE Release 10.1.3.3 includes the following fixes for Catalog Manager:

- Preferences. The dialog now allows 'browsing' to find a user's preferred browser.
- Create Report functionality. For import to Excel 2003, save to a file with a .tab extension.
- Search and replace.
- Renaming and deleting "special" folders (for example, _portal and _ibots) provides user warning.
- Hidden attributes can now be edited offline and online.
- Timestamps display.
- Sorting by type.
- Editing xml. Manual edits are not checked for validity.
- Synchronization of permissions between Catalog Manager and the online Manager Catalog feature.
- Renaming existing properties.
- Expanding and collapsing the tree view.
- Link handling.
- Table view - Owner and Attributes parameters have been added.

8.4 Upgrading Catalogs for EBA

You can use Catalog Manager to upgrade EBA ("EBA" stands for Enterprise Business Analytics) catalogs from Version 7.9.6.1 to 7.9.6.2. Catalog Manager does the following:

1. Compares the content in both the Baseline EBA Presentation Catalog and the Production Presentation Catalog with the content in the Latest EBA Presentation Catalog.

2. Merges any changes into the Production Presentation Catalog, which is why it is imperative that you use a copy of the Production catalog and perform the upgrade in a development or test environment so as to confirm the upgrade has worked properly.
3. Produces a list of upgrade differences, which you must resolve by indicating how you want the differences handled.

If the catalogs have conflicting content, then you can choose which catalog the content is taken from. The end result is a upgraded presentation catalog that contains the site-specific changes and new metadata.

You use a dialog to select the catalogs to include while upgrading an EBA catalog, as described in the following steps:

1. Make a backup copy of the Production Catalog, which is the catalog that is currently in use at your organization. Rename the directory to *catalog_name_old* and move it to a temporary location.
2. Start Catalog Manager in offline mode and open the copy of the Production Catalog.
3. From the Tools menu, select **Upgrade EBA Catalog**.
The Upgrade Customer Modified EBA Presentation Catalog dialog is displayed. The presentation catalog that is currently in use at your organization is automatically entered in the **Production Presentation Catalog** field.
4. In the **Baseline EBA Presentation Catalog** field, browse to locate the unmodified, read-only presentation catalog that you received with the Oracle BI EE release that you are upgrading from.
5. In the **Latest EBA Presentation Catalog** field, browse to locate the read-only presentation catalog for EBA Version 7.9.6.2 that is installed in the OracleBIData\web\catalog directory with the latest installation of Oracle BI EE.
6. Provide a log file to write the results of the process.
7. Click **OK**.

The upgrade process can take several minutes for a large catalog.

8. Resolve any upgrade differences as follows:
 - a. Review each conflict in the Resolve Conflicts dialog, selecting which version you want to keep. The default is to keep the version from the Production Catalog and to disregard conflicting changes from Oracle.
 - b. Click **OK**.

Review the log file that you selected in Step 6, which holds information about the merge process. This log file is written to the OracleBI\web\catalogmanager directory. If you see an error logged in the file, then the path in question had a problem that did not allow the merge mechanism to resolve the merge. No action was taken. To merge that particular item, go into the original presentation catalog and merge it manually.

9. Review the upgraded presentation catalog and if necessary, set permissions for objects.
10. Save the new presentation catalog.

8.5 Guidelines for Object Names

Each object in the catalog is stored in its own file. For example, an analysis called Analysis 1 is stored in a file named Analysis1. The object name that is visible to users, such as Analysis 1, is referred to as the logical object name.

The following list provides guidelines for object names:

- No restrictions exist on which characters are allowed in the logical name of an object in the catalog, provided that the characters are valid Unicode characters. The following are valid logical names:

```
Hello World
Profit / Loss
% Sales * $ Cost ~~ $ "Expense"?
```

- The length of the logical object name must not exceed 256 Unicode characters. For more information on Unicode, see [Section 8.6, "File System Guidelines for Catalogs."](#)
- The length of the logical path name for an object must not exceed 16000 Unicode characters.
- The number of directory segments in a logical path name for an object must be not exceed 255 segments.

For example, a directory with a name such as
/n1/n2/n3/n4/.../n253/n254/n255 is acceptable, while a name such as
/n1/n2/n3/n4/.../n254/n255/n256 is unacceptable.

- When you pass the path name of an object using SOAP, you must escape the following characters:

```
Forward slash (/)
Backward slash (\)
Tilde (~)
Asterisk (*)
Question mark (?)
```

The following logical path names are all valid:

```
/shared/test/Hello World
/shared/test/Profit \ / Loss
/shared/test/% Sales \* $ Cost \~\~ $ "Expense" \?
```

Use care when building a catalog path. It is very common to see code that assumes the forward slash (/) is always a path separator. Always verify your path code with an object name such as "Profit / Loss".

- When you pass a catalog search filter using SOAP, you must escape the following characters:

```
Forward slash (/)
Backward slash (\)
Tilde (~)
Asterisk (*)
Question mark (?)
Caret (^)
Dollar sign ($)
```

The following search filters are all valid:

```
Hello World
```

```
Profit \ / Loss
% Sales \* \ $ Cost \~\~ \ $ "Expense" \?
```

8.6 File System Guidelines for Catalogs

This section describes the following guidelines for working with objects in catalogs in file systems:

- [Section 8.6.1, "Handling Users of the Catalog"](#)
- [Section 8.6.2, "Handling Heterogeneous Nodes"](#)
- [Section 8.6.3, "Handling Catalog Files on Various Platforms"](#)
- [Section 8.6.4, "Known Issues with Catalog Files"](#)

8.6.1 Handling Users of the Catalog

The catalog is designed to scale to thousands of concurrent users. To achieve this scaling, the catalog adheres to the following guidelines:

- The average user typically only reads from the catalog and rarely, if ever, writes to it. In Release 11g, each user is constantly and automatically updating his or her Most Recently Used file, but each user's "read" operations still far outweigh the user's "writes" operations. Therefore, the read-to-write ratio is typically at least 100 to 1.
- While a locking mechanism guarantees that only one user can write to an object at a time, it is rare for multiple users to attempt to write simultaneously to the same object. A feature called "lazy locking" allows users to continue reading an object even when another user is updating that object.
- Modern file systems cache "small" files directly inside the directory record, such that reading any information on a directory simultaneously loads all small files directly into the operating system's memory cache. Therefore, it is good practice to keep files in the catalog "small," especially the frequently "read" .atr metadata files. When these metadata files remain small, then all the .atr files in a directory are loaded into memory with one physical hard disk read. Every file that exceeds the "small" threshold adds another physical hard disk read, which can cause a 100% degradation for each large file. In other words, use care when considering storing arbitrary "Properties" in .atr files.
- Reading an object's .atr metadata file using NFS is far slower than reading it directly from a local disk. For this reason, Presentation Services additionally caches all .atr files internally. This cache can become briefly "stale" when another node in the cluster writes data to the file that is newer than the data that is cached by the current node. Therefore, all nodes are refreshed according to the setting of the MaxAgeMinutes element in the instanceconfig.xml, whose default for a cluster is 5 minutes. This default setting commonly achieves the best trade-off between the possibility of stale data and the known performance impact. (The default for an environment without clusters is 60 minutes.)

8.6.2 Handling Heterogeneous Nodes

To allow heterogeneous nodes in a cluster, the catalog adheres to the following guidelines:

- The maximum length for the name of an object on disk is 256 bytes, which is 64 Unicode characters. The logical name is restricted to 256 Unicode characters. To adhere to this restriction, logical names greater than 32 characters are hashed.
- The maximum length for the name of a path on disk is 32KB, which is 8000 Unicode characters. The logical path is restricted to 16000 Unicode characters.
- All path names on disk are all lowercase. The logical path name allows mixed case, but is still case-insensitive.
- Certain characters are not allowed for path names on disk, while the logical path name allows all characters. For example, Windows systems disallow certain characters such as the colon (:), so those characters are mapped using standard HTML escape sequences. For example, the period character (.) becomes "%2e".
- Certain file names are not allowed on disk, while the logical object name has no restrictions. For example, Windows systems disallow certain file names such as COM, so those names are mapped using standard HTML escape sequences. For example, "com" becomes "co%6d".

8.6.3 Handling Catalog Files on Various Platforms

Keep the following points in mind when handling catalog files on various platforms:

- For UNIX Platforms: UNIX kernels must commonly be configured to allow more than 4000 subdirectories per directory. See Chapter 18 for information on the HashUserHomeDirectories element.
- For Windows Platforms:

When users want to navigate catalog files using a tool such as Microsoft Windows Explorer, then they want the catalog structure based on a short path name such as `c:/obi/demo`, rather than the long default path name. Note that such navigation is not recommended.

- FAT is not supported, and NTFS is required.
- Performance on Windows platforms degrades noticeably when more than 8000 files exist in a single directory. Because each catalog object has two files (the data file and the .atr metadata file), it is strongly recommended that you not store more than 4000 catalog objects in a single directory. See Chapter 18 for information on the HashUserHomeDirectories element.
- Windows Explorer does not handle long path names properly, and it is recommended to not Windows Explorer to navigate the internal structure of the catalog. While the file system can handle path names as large as 32KB and Presentation Services is not negatively affected, you cannot use Windows Explorer with any path name that is longer than approximately 2KB.

Because a single Unicode character can require as many as 4 bytes, you might be unable to use Windows Explorer with path names of only 500 Unicode characters. This limitation does not affect Presentation Services. Because of this limitation, place the catalog in a top-level directory, such as `c:\mycatalog\sales`.

8.6.4 Known Issues with Catalog Files

The following issues are known when working with catalog files:

- Locking across NFS systems is difficult, but Presentation Services provides an effective locking mechanism in recent patches. Obtain key patches to update older versions of Oracle BI EE as necessary.

For more information, see [Section 13.2, "Validating the Catalog."](#)

- Various third-party FTP programs have issues handling '%' escape sequences, which often results in a renamed file that is doubly escaped. For example, a file that is named sa%2epaint (whose logical name is SA.Paint) is incorrectly renamed to sa%252epaint (whose logical name is SA%2ePaint).

Avoid using an FTP program directly against a catalog. Instead, download and use the 7-Zip utility to compress the catalog files, then use an FTP program to transfer the resulting compressed file.

8.7 Limitations and Workarounds in Catalog Manager

Ensure that you keep the following items in mind when working with Catalog Manager:

- Attempts to copy, archive, or drag files from the /system/security directory cannot be performed within the Catalog Manager as these functions should be performed elsewhere.
- While working in the online mode, catalog contents cannot be pasted into or out of a read-only folder. To achieve this result, turn-off the read-only property of the folder tree before copying, then re-apply read-only after pasting.
- Items with special characters in their names might not be displayed correctly.
- Even if a resize indicator is not shown, Catalog Manager panes might still be resizable.
- After exiting the Catalog Manager after working in offline mode, its Java engine might continue to display in the Windows Task Manager.
- The Catalog Manager cannot be run from the Windows Start > Run menu. To start the Catalog Manager using Windows, double-click the executable within Windows Explorer or provide the full path from the command line.
- Attempting to use the Upgrade Catalog feature on an empty catalog might cause an error. If you see an error, then restart the Catalog Manager.
- If the Catalog Manager becomes unusable after closing both the tree and table panes, then restart the Catalog Manager.
- The Catalog Manager's Status Log provided little value and has been removed.
- Archives created using compression through the online Manage Catalog feature found in Answers cannot be loaded using the Catalog Manager application's unarchive facility.
- Archives created in 10.1.3.2.x are not binary compatible with 10.1.3.3. Migrations can be made using unarchived Presentation Catalogs.

General New Features in Release 10.1.3.3.1

Oracle BI EE Release 10.1.3.3.1 introduces three new BI Server features. This chapter describes the following new features:

- [Section 9.1, "Oracle Business Intelligence Installer Deploys BI Components into One OC4J Container"](#)
- [Section 9.2, "DataDirect Connect ODBC Drivers for Linux and UNIX Operating Systems Bundled with Release 10.1.3.3.1"](#)
- [Section 9.3, "Support for Charts in Arabic, Thai, and Hebrew"](#)

9.1 Oracle Business Intelligence Installer Deploys BI Components into One OC4J Container

In 10.1.3.x releases before 10.1.3.3.1, during an Advanced installation, the Oracle BI Installer created the following OC4J containers and deployed BI components in them:

- bianalytics - for the BI Presentation Services Plug-in component
- bipublisher - for the BI Publisher component
- bijmx - for the BI Systems Management component
- biooffice - for the Oracle BI Office Server component (in Oracle BI EE version 10.1.3.3)

With Release 10.1.3.3.1, during an Advanced installation, the Oracle BI Installer deploys the BI components into one OC4J container called "home". The memory footprint for the OC4J instance is between 512 -1024 MB.

9.1.1 Upgrading to 10.1.3.3.1 from a BIEE 10.1.3.x Release Before 10.1.3.3.1

On upgrade to Release 10.1.3.3.1, if the "Keep User Modified Configurations" upgrade option is selected on the BI Installer screen, the old OC4J containers for BI are deleted and the BI components are deployed into the OC4J container named "home". The configuration files for the BI component applications deployed in the individual BI OC4J containers are migrated into the deployed application in the OC4J instance named "home".

The configuration files that are migrated are:

```
$ORACLE_HOME/j2ee/bianalytics/applications/analytics/analytics/WEB-INF/web.xml  
to  
$ORACLE_HOME/j2ee/home/applications/analytics/analytics/WEB-INF/web.xml  
and
```

ORACLE_HOME/j2ee/biiooffice/applications/analytics/analytics/WEB-INF/biooffice.xml
to

\$ORACLE_HOME/j2ee/home/applications/analytics/analytics/WEB-INF/biooffice.xml

On upgrade to Release 10.1.3.3.1, the old OC4J containers for BI are deleted if the "Reset Configurations" upgrade option is selected on the BI Installer screen. The BI components are deployed in the OC4J instance named "home". The configuration files web.xml for bianalytics and biooffice.xml from the previous release are not preserved.

Note: If you have deployed any custom applications on any of the BI OC4J containers, then you must make a backup of the custom applications and redeploy them into a different OC4J container before performing an upgrade to Release 10.1.3.3.1.

9.2 DataDirect Connect ODBC Drivers for Linux and UNIX Operating Systems Bundled with Release 10.1.3.3.1

Release 10.1.3.3.1 provides DataDirect Connect ODBC drivers and driver managers version 5.3 for Linux and UNIX operating systems for connectivity to Informix, MS SQL Server, and Sybase ASE databases. 32-bit and 64-bit DataDirect 5.3 drivers are available on AIX, Solaris, and HP-UX Itanium. 32-bit drivers are available for Linux and HP-UX PA-RISC.

The Release 10.1.3.3.1 installer installs the DataDirect Connect ODBC drivers in the following location:

\$OracleBI/odbc

The 32-bit drivers are installed in \$OracleBI/odbc/lib and 64-bit drivers are installed in \$OracleBI/odbc/lib64.

Note: Communication between database clients and servers is typically independent of the widths and data paths. In other words, the 32-bit database drivers communicate with 64-bit database servers, and vice versa.

Refer to *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence 10.1.3.3* for further information on supported operating systems and databases for the DataDirect Connect ODBC drivers.

9.2.1 Using the DataDirect Connect ODBC Drivers

The following examples show sample ODBC.ini file entries for Microsoft SQL Server, Sybase ASE, and Informix data sources.

9.2.1.1 For Connectivity to Microsoft SQL Server Database

The name of the DataDirect ODBC driver file to connect to an MS SQL Server database is SEMSSS23.so (SEMS23.sl on HP-UX PA-RISC). For supported versions of MS SQL Server see *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence 10.1.3.3*.

On Linux, AIX, Solaris, HP-UX PA-RISC and HP-UX Itanium operating systems, the 32-bit driver is located in \$OracleBI/odbc/lib. On AIX, Solaris and HP-UX Itanium operating systems, the 64-bit driver is located in \$OracleBI/odbc/lib64.

To use the DataDirect ODBC Driver to connect to MS SQL Server:

1. Make the following entries in the `odbc.ini` file. This file is located in `$OracleBI/setup`. Set the Driver parameter to the file name and location of the DataDirect driver for MS SQL Server. In the following example, the Driver parameter is set to the 64-bit DataDirect driver.

Example 9–1 Sample File for Microsoft SQL Server

```
[SQLSERVER_DB
Driver=/usr/OracleBI/odbc/lib64/SEmsss23.so
Description=DataDirect 5.1 SQL Server Wire Protocol
Address=111.111.111.111,1433
AlternateServers=
AnsiNPW=Yes
ConnectionRetryCount=0
ConnectionRetryDelay=3
Database=
LoadBalancing=0
LogonID=
Password=
QuoteID=No
ReportCodePageConversionErrors=0
```

2. In the `$OracleBI/server/Config/DBFeatures.INI` file, for the appropriate SQL Server sections, turn the IDENTIFIER_QUOTE_CHAR parameter off by setting the following:

```
IDENTIFIER_QUOTE_CHAR = ";
```

The default is:

```
IDENTIFIER_QUOTE_CHAR = "";
```

9.2.1.2 For Connectivity to Sybase ASE Database

The name of the DataDirect ODBC driver file to connect to a Sybase ASE database is `SEase23.so`. For supported versions of Sybase ASE see *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence 10.1.3.3*.

On Linux, AIX, Solaris, HP-UX PA-RISC and HP-UX Itanium operating systems, the 32-bit driver is located in `$OracleBI/odbc/lib`. On AIX, Solaris and HP-UX Itanium operating systems, the 64-bit driver is located in `$OracleBI/odbc/lib64`.

To use the DataDirect ODBC Driver to connect to Sybase ASE, make the following entries in the `odbc.ini` file. This file is located in `$OracleBI/setup`. Set the Driver parameter to the file name and location of the DataDirect driver for Sybase ASE. In the following example, the Driver parameter is set to the 64-bit DataDirect driver.

Example 9–2 Sample File for Sybase ASE Data Source

```
[SybaseASE_DB
Driver=/usr/OracleBI/odbc/lib64/SEase23.so
Description=DataDirect 5.3 Sybase Wire Protocol
AlternateServers=
ApplicationName=
ApplicationUsingThreads=1
ArraySize=50
AuthenticationMethod=0
Charset=
ConnectionRetryCount=0
ConnectionRetryDelay=3
```

```
CursorCacheSize=1
Database=Northwind
DefaultLongDataBuffLen=1024
EnableDescribeParam=0
EnableQuotedIdentifiers=0
EncryptionMethod=0
GSSClient=native
HostNameInCertificate=
InitializationString=
Language=
LoadBalancing=0
LogonID=
NetworkAddress=111.111.111.111,5005
OptimizePrepare=1
PacketSize=0
Password=
RaiseErrorPositionBehavior=0
ReportCodePageConversionErrors=0
SelectMethod=0
ServicePrincipalName=
TruncateTimeTypeFractions=0
TrustStore=
TrustStorePassword=
ValidateServerCertificate=1
WorkStationID=
```

9.2.1.3 For Connectivity to Informix Database

The name of the DataDirect ODBC driver file to connect to an Informix database is SEifcl23.so. For supported versions of Informix see *Systems Requirements and Supported Platforms Guide for Oracle Business Intelligence 10.1.3.3*.

On Linux, AIX, Solaris, HP-UX PA-RISC and HP-UX Itanium operating systems, the 32-bit driver is located in \$OracleBI/odbc/lib. On AIX, Solaris and HP-UX Itanium operating systems, the 64-bit driver is located in \$OracleBI/odbc/lib64.

To use the DataDirect ODBC Driver to connect to Informix, make the following entries in the odbc.ini file. This file is located in \$OracleBI/setup. Set the **Driver** parameter to the file name and location of the DataDirect driver for Informix. In the following example, the Driver parameter is set to the 64-bit DataDirect driver.

Example 9-3 Sample File for Informix Data Source

```
[Informix_DB]
Driver=/usr/OracleBI/odbc/lib64/SEifcl23.so
Description=DataDirect Informix Wire Protocol
AlternateServers=
ApplicationUsingThreads=1
CancelDetectInterval=0
ConnectionRetryCount=0
ConnectionRetryDelay=3
Database=
HostName=111.111.111.111
LoadBalancing=0
LogonID=informix
Password=inform_10
PortNumber=1526
ReportCodePageConversionErrors=0
ServerName=
TrimBlankFromIndexName=1
```

9.3 Support for Charts in Arabic, Thai, and Hebrew

Release 10.1.3.3.1 offers support for the Arabic, Thai, and Hebrew languages. Due to a limitation in the charting engine that is used in Oracle BI EE, bidirectional and Thai language characters do not render properly in charts.

To overcome this limitation and enable charting in Oracle BI EE for Arabic, Thai and Hebrew languages, you can use Oracle BI Publisher charting capabilities. BI Publisher charts are based on the Oracle BI Beans technology. A BI Publisher report that contains a chart might be embedded in an Oracle BI Interactive Dashboard to meet charting requirements.

To implement this solution, perform the following steps described below:

- [Integrate the Data Sources](#)
- [Create a Report and Layout in BI Publisher](#)
- [Embed the BI Publisher Report in a Dashboard](#)

9.3.1 Prerequisites

This procedure assumes that you have previously performed the following tasks:

- Set up users in Oracle BI EE and Oracle BI Publisher with appropriate privileges.

You must integrate the user security between Oracle Business Intelligence and Oracle BI Publisher using an available security model. Within Oracle BI Presentation Services the user must have privileges to update Dashboards. Within Oracle BI Publisher users must be granted the BI Publisher Developer Role.

For more information on setting up security in Oracle BI, refer to "Security in Oracle BI" in *Oracle Business Intelligence Server Administration Guide* and "Managing Oracle Presentation Services Security" in *Oracle Business Intelligence Presentation Services Administration Guide*. For information on setting up security in Oracle BI Publisher, refer to "Defining a Security Model" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

- Created an Answers request that contains the data to display in a chart.
- Installed the BI Publisher Template Builder for Microsoft Word.

The Oracle BI Publisher Template Builder for Microsoft Word Add-in can be installed from the BI Publisher application for users with the appropriate roles assigned (that is, BI Publisher Administrator, BI Publisher Developer, or BI Publisher Template Designer). The download link is displayed on the **Developer Tools** region. See the topic "Installing Oracle BI Publisher Desktop" in *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

9.3.2 Integrate the Data Sources

You must set up the Oracle BI Server as a data source in BI Publisher. To use an Answers request as a data source, you must also set up integration with Oracle BI Presentation Services.

9.3.2.1 Adding the Oracle BI Server as a Data Source

If you included BI Publisher in the Oracle BI EE installation, then the Oracle BI Server is automatically configured as a data source in BI Publisher. If you did not include BI Publisher during the Oracle BI EE installation, then you must manually add the BI Server as a data source using BI Publisher's Admin interface. For detailed steps, see the

topic: "Adding the Oracle BI Server as a JDBC Data Source" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

9.3.2.2 Integrating with Presentation Services

By setting up integration with Oracle BI Presentation Services, you enable connection to Oracle BI Answers requests as data sources for your reports.

The BI Publisher Admin interface provides a specific page for integration with Oracle BI Presentation Services. Navigate to this page from the Admin page by selecting **Oracle BI Presentation Services** under the **Integration** subheading.

If you included BI Publisher in the Oracle BI EE installation, then the Oracle BI installer sets the Presentation Services host name, port, and URL values. However, you must manually enter the Presentation Services user name and password on the Oracle BI Presentation Services integration page.

9.3.3 Create a Report and Layout in BI Publisher

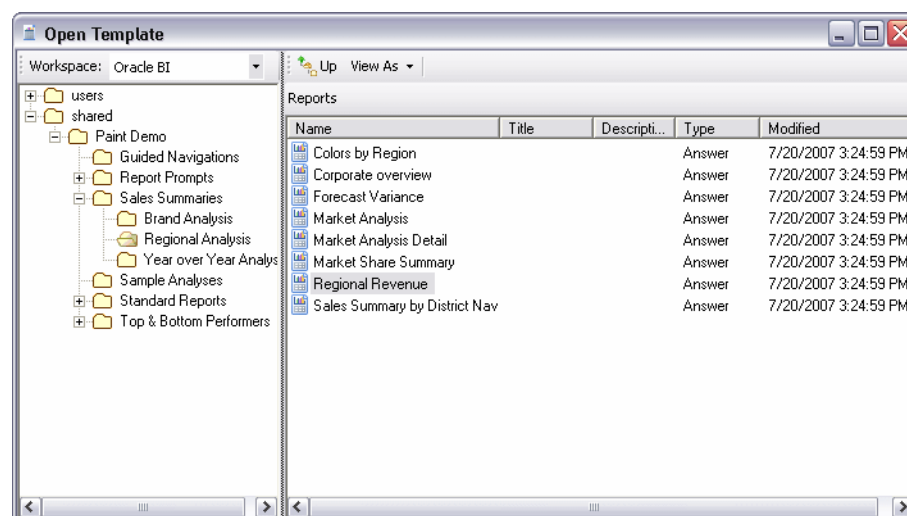
Following are summary steps for creating a report and layout in BI Publisher. For detailed description refer to the following sources:

For detailed steps to create a report in BI Publisher, see the chapter "Creating a New Report" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

For information on using the Template Builder for Microsoft Word, see the Template Builder online help.

1. Open Microsoft Word (with the Template Builder for Word installed).
2. From the Oracle BI Publisher menu, select **Log On** and enter appropriate credentials (you must have either the Administrator, Report Designer, or Developer role assigned).
3. From the Open Template dialog, select Oracle BI from the **Workspace** list.
4. Navigate to your Answers request and double-click the request name.

Figure 9–1 Open Template Dialog



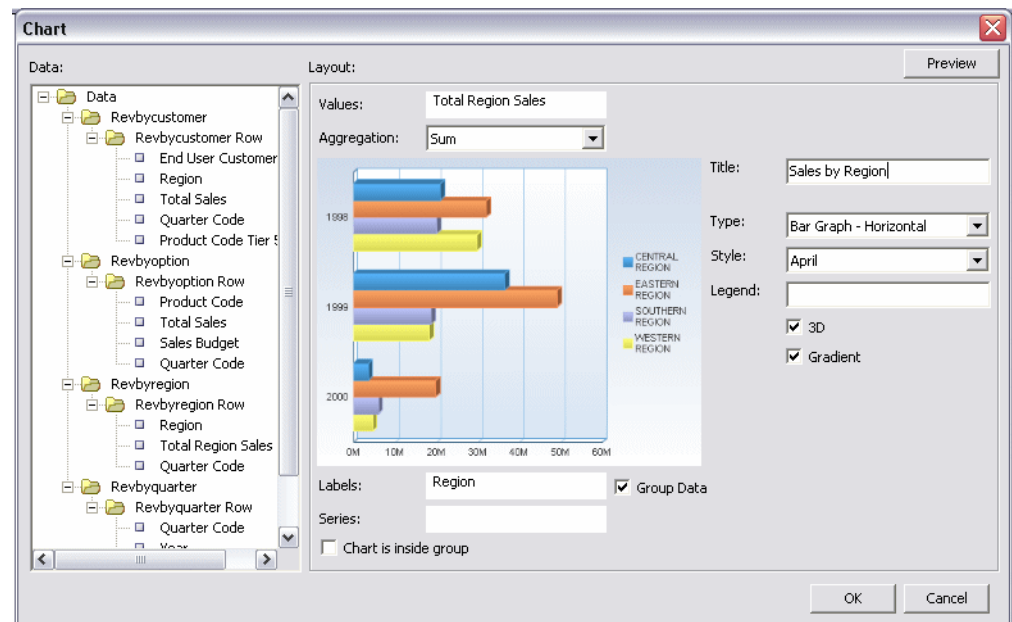
5. You are prompted to **Save as Oracle BI Publisher Report**. BI Publisher creates a report definition that uses the Answers request as the data source. The template

that you design is saved as part of this BI Publisher report definition. Select the folder from the BI Publisher report repository.

After you save, the Template Builder loads the data and returns you to a blank Microsoft Word page.

6. To insert a chart, launch the Chart dialog from the Oracle BI Publisher menu by selecting **Insert**, then **Chart**. Figure 9–2 shows the Oracle BI Publisher Template Builder chart dialog.

Figure 9–2 Template Builder Chart Dialog



7. After inserting the chart, upload the template to the BI Publisher report repository. From the Oracle BI Publisher menu, select **Upload Template As**. Enter a name for the template.

The report is available as a BI Publisher object that can be added to a dashboard.

9.3.4 Embed the BI Publisher Report in a Dashboard

The following procedure describes how to embed the BI Publisher report in a Dashboard. For more detailed instructions and options, refer to the topic "Adding an Oracle BI Publisher Report to an Oracle BI Interactive Dashboard Page" in *Oracle Business Intelligence Answers, Delivers, and Interactive Dashboards User Guide*.

Note: In Release 10.1.3.3.1, there is no longer a limitation of one BI Publisher Report per dashboard.

1. In the dashboard page, click the **Page Options** button, and click **Edit Dashboard**.
2. From the selection pane, drag and drop the BI Publisher Report object from the Dashboard Objects area into a section on the dashboard page.
3. Click the newly placed BI Publisher report object's **Properties** button.
4. In the BI Publisher Report Properties dialog, browse for and select the Oracle BI Publisher report.

5. Click **Save** to return to the dashboard and view your added report.

9.3.5 Supported Graph Types

The following graph types are supported in BI Publisher:

- Area
- Horizontal Bar
- Bubble
- Vertical Bar
- Line
- Line Bar Combo
- Pareto
- Pie
- Radar
- Scatter
- Step

9.3.6 Known Limitations

The BI Publisher chart is static.

The locale information from the BI Server is not inherited by BI Publisher. BI Publisher charts embedded in the dashboard take the locale setting from BI Publisher. To ensure consistency, set the same locale for both applications.

General New Features in Release 10.1.3.3.2

Oracle BI EE Release 10.1.3.3.2 introduces the following new BI Server features:

- [Section 10.1, "Support for Hyperion Essbase as a Data Source"](#)
- [Section 10.2, "Using BI EE OC4J Components with IIS Through the IIS Plug-In"](#)

10.1 Support for Hyperion Essbase as a Data Source

Oracle BI EE Release 10.1.3.3.2 supports Hyperion Essbase as a physical data source. This topic describes the process to import Essbase structures into the Oracle BI Server metadata repository and usage guidelines and limitations.

10.1.1 Supported Versions of Hyperion Essbase with Oracle BI EE 10.1.3.3.2

At the time of this release, Hyperion Essbase version 7.1.6 and later can be used as a data source with Oracle BI EE Release 10.1.3.3.2. For the most current information regarding supported versions, see *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition*.

10.1.2 Enabling the Use of Hyperion Essbase as a Data Source

This section describes the steps required to enable Essbase as a data source.

10.1.2.1 Essbase Client Libraries

Oracle BI Server connectivity to Essbase is through the Essbase client libraries. The client libraries must be installed on the BI Server. See *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition* document for the supported versions of the Essbase Client for connectivity to the BI Server.

10.1.2.2 BI Server Configuration for UNIX and Linux Platforms

Perform the following configuration steps to access the Essbase client libraries through the BI Server on UNIX and Linux Platforms:

10.1.2.2.1 Solaris: Oracle BI 64-Bit Mode

1. Define ARBORPATH = <Essbase Client installation folder>

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to LD_LIBRARY_PATH:

```
LD_LIBRARY_PATH_64=$LD_LIBRARY_PATH_64:<Essbase Client Libraries folder>
```

For example:

```
LD_LIBRARY_PATH_64=$LD_LIBRARY_PATH_64:$ARBORPATH/bin
export LD_LIBRARY_PATH_64
```

10.1.2.2.2 Solaris: Oracle BI 32-Bit Mode

1. Define ARBORPATH = <Essbase Client installation folder>

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to LD_LIBRARY_PATH:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<Essbase Client Libraries folder>
```

For example:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH$ARBORPATH/bin
export LD_LIBRARY_PATH
```

10.1.2.2.3 HP-UX PARISC: Oracle BI 32-Bit Mode

1. Define ARBORPATH = <Essbase Client installation folder>

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to SHLIB_PATH:

```
SHLIB_PATH=$SHLIB_PATH:<Essbase Client Libraries folder>
```

For example:

```
SHLIB_PATH=$SHLIB_PATH$ARBORPATH/bin
export SHLIB_PATH
```

10.1.2.2.4 HP-UX Itanium: Oracle BI 64-Bit Mode

1. Define ARBORPATH = <Essbase Client installation folder>

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to SHLIB_PATH:

```
SHLIB_PATH=$SHLIB_PATH:<Essbase Client Libraries folder>
```

For example:

```
SHLIB_PATH=$SHLIB_PATH:$ARBORPATH/bin
export SHLIB_PATH
```

3. Define ESSLANG and LANG

For example:

```
ESSLANG=English_UnitedStates.UTF-8@Binary
export ESSLANG
LANG=en_US.utf8
export LANG
```

4. Comment out the following three lines from the NQSCONFIG.INI file:

```
[ GENERAL ]
// Localization/Internationalization parameters.
LOCALE="English-usa";
SORT_ORDER_LOCALE="English-usa";
SORT_TYPE="binary";
```

10.1.2.2.5 AIX Oracle BI 32-Bit and 64-Bit Modes

1. Define ARBORPATH = *<Essbase Client installation folder>*

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to LIBPATH:

```
LIBPATH=$LIBPATH:<Essbase Client Libraries folder>
```

For example:

```
LIBPATH=$LIBPATH:$ARBORPATH/bin
export LIBPATH
```

10.1.2.2.6 Linux: Oracle BI 32-Bit Mode

1. Define ARBORPATH = *<Essbase Client installation folder>*

For example:

```
ARBORPATH=/export/home/Hyperion/AnalyticServicesClient
export ARBORPATH
```

2. Add the Essbase Client Libraries folder to LD_LIBRARY_PATH:

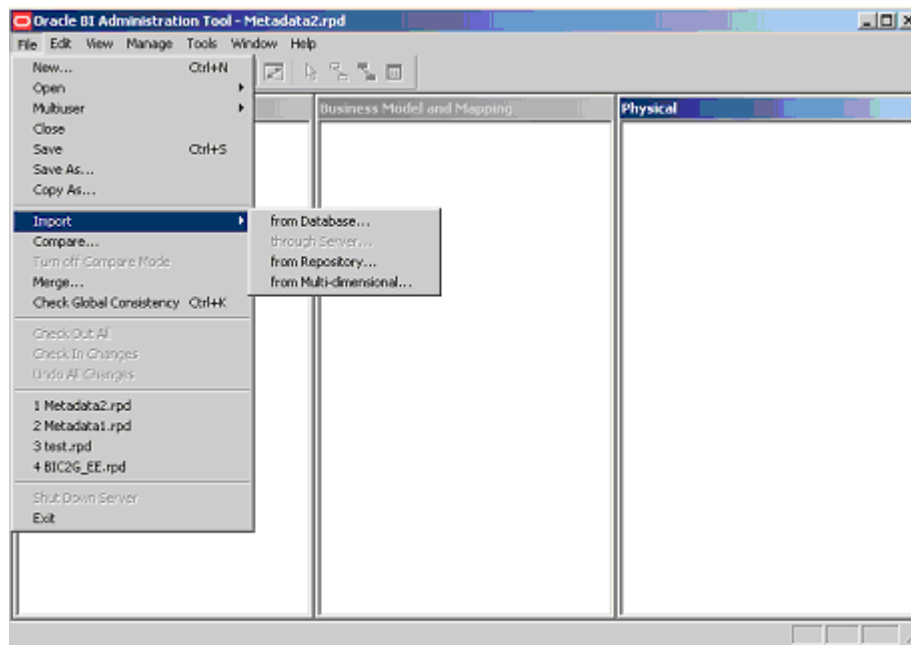
```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:<Essbase Client Libraries folder>
```

For example:

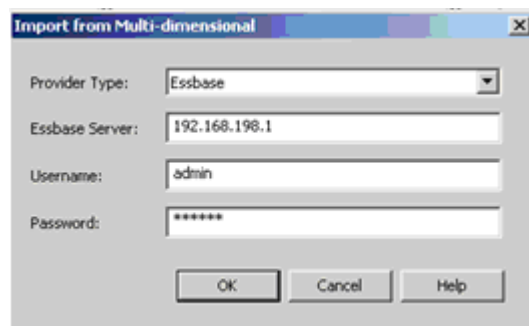
```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ARBORPATH/bin
export LD_LIBRARY_PATH
```

10.1.2.3 Import Process

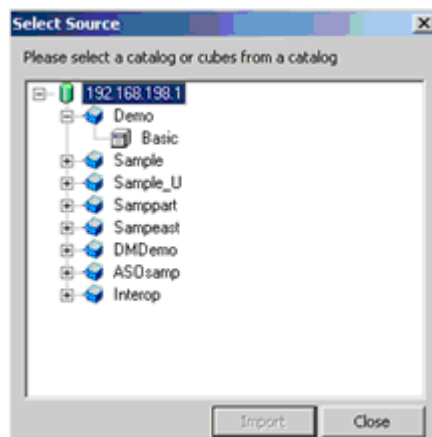
Essbase cube structures are imported through the "Import from Multi-dimensional" option in the Oracle BI Administration tool, shown in the following figure:



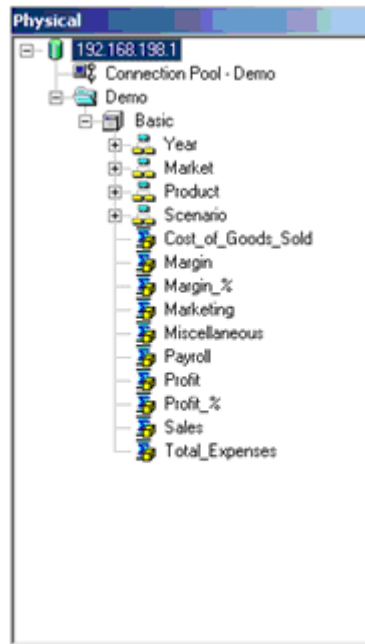
In the "Import from Multi-Dimensional" dialog, select Essbase as the provider type. Input the server and authentication information.



After selecting "OK", the list of Applications and Cubes available on the Essbase Server are presented as shown in the following figure:



Select the Applications or Cubes to import. Note that Applications are mapped as Physical Catalogs. The following figure shows an example of the physical layer representation:



Once the physical cubes are imported, they can be dragged to the Business Model and Mapping Layer to automatically create a Business Model that includes associated dimensions. The Business Model can now be customized to take advantage of Oracle BI Server capabilities such as aggregate and fragment navigation.

10.1.3 Guidelines and Limitations

This section describes some unique aspects to Essbase connectivity that the administrator or end user should be aware of when configuring and using Essbase as a data source.

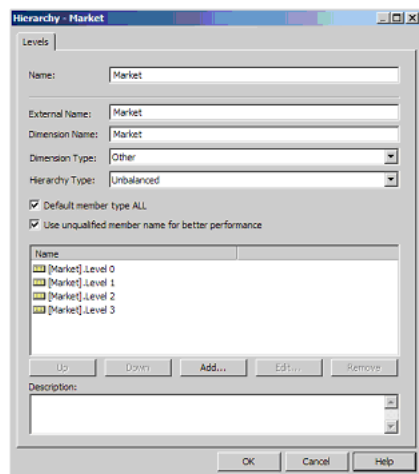
10.1.3.1 Measures

The default Aggregation Rule for imported measures is set to Aggr_External. It is recommended that you change this setting to an explicit Aggregation Rule if the rule is known for the given measure and a corresponding rule is available for the BI Server. This allows the BI Server to compute the aggregate when necessary.

Essbase supports the concept of measure hierarchies. This support enables end users to drill from a measure to components that comprise the measure, for example, profit drills to revenue and costs. Currently Oracle BI EE does not support measure hierarchies. Oracle BI EE imports measures as a flat list of children to the cube itself, independent of the measure hierarchy.

10.1.3.2 Unique Member Names

When member names (including aliases) are unique in a given hierarchy, the BI Server can take advantage of specific MDX syntax to optimize performance. To enable this capability, select the "Use unqualified member name for better performance" box in the Hierarchy dialog as illustrated in the following figure.



The import process cannot identify that member names are unique for a given hierarchy, so it is the responsibility of the Administrator to confirm uniqueness. Note that query errors might result if a hierarchy is specified as having unique members when it does not.

10.1.3.3 Time Series Functions

The BI Server Time Series functions, "Ago" and "ToDate", are supported for use with Essbase.

10.1.3.4 Evaluate Function for MDX Sources

Oracle BI EE Release 10.1.3.3 introduced Evaluate functions to provide the ability to leverage unique source-specific functions. This capability was limited to relational sources. In Release 10.1.3.2, you can leverage Evaluate functions for Essbase. Support for Evaluate does not currently extend across all multi-dimensional sources.

Sample use cases are described in the following sections. Consider the following examples using the EVALUATE_AGGREGATE and EVALUATE functions. Note that expressions are applied to columns in the Logical Table Source that refers to the physical cube.

EVALUATE_AGGREGATE is used to implement custom aggregations. For example, you might want to compare overall regional profit to profits for the top three products in the region. A new measure can be defined to represent the profits for top three products resulting in the logical SQL statement:

```
Select Region, Profit, EVALUATE_AGGREGATE('SUM(TopCount(%1.members, 3, %2), %3)',
Products, Profit, Profit) Top_3_prod_Profit From SampleBasic
```

The BI Server generates the following expression for the custom aggregation:

```
member [Measures].[MS1] as
'SUM(Topcount([Product].Generations(6).members,3,[Measures].[Profit]),[Measures].[
Profit])'
```

Similarly, use the EVALUATE function to implement scalar functions that are computed post-aggregation. EVALUATE changes the grain of the query, if its definition makes explicit references to dimensions (or attributes) that are not in the query.

For example, to see the Profits for the top five products ranked by Sales sold in a Region, after creating the applicable measure, the resulting Logical SQL statement is as follows:

```
Select Region, EVALUATE('TopCount(%1.members, 5, %2)' as VARCHAR(20), Products,
Sales), Profits
From SampleBasic
```

The BI Server generates the following expression to retrieve the top five products:

```
set [Evaluate0] as
'{Topcount([Product].Generations(6).members,5,[Measures].[Sales]) }'
```

10.1.3.5 User-Defined Attributes

Essbase supports the concept of user-defined attributes (UDAs). A UDA is essentially any arbitrary textual string that can be associated with any member from a dimension. A member can have multiple strings associated to it. Oracle BI EE suppresses UDAs upon import.

Users can still query using UDAs by leveraging the EVALUATE function. Consider the following example where "Major Market" is a UDA:

After creating the applicable column, the resulting logical SQL statement is as follows:

```
Select EVALUATE('FILTER( %1.Members, isUDA(%2.Dimension, "Major Market"))',
State, State), Sales
From SampleBasic
```

The BI Server generates the following expression for the custom column:

```
set [Evaluate0] as
'{FILTER([Customer].Generations(3).members,isUDA([Customer].Generations(3).Dimensi
on,"Major Market")) }'
```

10.1.3.6 Substitution Variables

Essbase substitution variables are automatically retrieved and populated into corresponding BI Server dynamic system variables.

Depending on the scope of the Essbase variable, the naming convention for the BI Server variable is as described below.

Server instance scope

<server name>:<var name>

Application scope

<server name>:<app name>:<var name>

Cube scope

<server name>:<app name>:<cube name>:<var name>

The refresh interval should be set appropriately to reflect anticipated update cycles for Essbase variables.

10.1.3.7 Attribute Dimensions

Member attributes are not automatically associated to corresponding dimensions and levels during the import process. To manually create the association, the Oracle BI Administrator can choose one of the following:

- Map the member attribute to the appropriate Logical Table in the Business Model Layer
- Include the attribute in the appropriate Presentation Table in the Presentation Layer

10.1.3.8 Unbalanced Hierarchies

Oracle BI EE Release 10.1.3.3.2 supports unbalanced hierarchies for Essbase sources. The following is an illustration of how unbalanced hierarchies are represented to end users in the Answers interface.

Given a selection of Country, Region as shown in the following image:



Title				
Table				
Country	Region			
	El Salvador			
Central America	Honduras			
	Panama			
	Central			
US	East			
	South			
	West			

Now add "State" to the query. For "Central America," Regions do not have States.

The resulting Answers request is shown in the following image:

Title		
Table		
Country	Region	State
Central America	El Salvador	
	Honduras	
	Panama	
US	Central	Colorado
		Illinois
		Iowa
		Missouri
		Ohio
		Wisconsin
	East	Connecticut
		Florida
		Massachusetts
		New Hampshire
		New York
	South	Louisiana
		New Mexico
		Oklahoma
		Texas
	West	California
		Nevada
		Oregon
		Utah
		Washington

Note that for the unbalanced branch, for columns below the leaf level, nulls are displayed.

10.2 Using BI EE OC4J Components with IIS Through the IIS Plug-In

There are two components of the Oracle BI EE installation that require Oracle Containers for Java (OC4J) and do not run natively on Microsoft's Internet Information Server (IIS). These components are:

- Oracle BI Publisher
- Oracle BI Office Server

IIS can be configured as a listener for OC4J. This is accomplished through an IIS proxy plug-in that is provided with the Oracle BI EE installation files. When configured, the requests are routed from IIS to OC4J so that it seems to the user that everything is being executed by IIS.

10.2.1 Prerequisites

Following are the prerequisites to use the IIS plug-in:

- Certified Windows Operating System as documented in *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition (SRSP)*.
- Certified version of IIS as documented in the SRSP.
- Oracle BI EE Release 10.1.3.3.1 or later, installed and running with default configuration.

10.2.2 Configuration Steps

The Oracle Application Server Proxy Plug-in is called `oracle_proxy.dll` and is located in the Oracle BI EE installation files. The following steps describe how to configure to use the IIS plug-in:

1. In the Oracle BI EE install files, locate `oracle_proxy.dll`. The navigation path is as follows:

```
<CD-Drive>\Server\Oracle_Business_
Intelligence\oc4jproxy\oracle_proxy.dll
```

2. Create a folder on an accessible drive, for example: `c:\proxy`. Copy `oracle_proxy.dll` to this folder.
3. In the same folder, create a configuration file called "`proxy.conf`". This configuration file is for the OracleAS Proxy Plug-in which is used to define the proxy server behavior. Following is a sample configuration file:

```
# Server names that the proxy plug-in recognizes.
oproxy.serverlist=ias1

# Hostname to use when communicating with a specific server.
oproxy.ias1.hostname=hostname.domain

# Port to use when communicating with a specific server.
oproxy.ias1.port=9704

# Description of URL(s) that are redirected to this server.
oproxy.ias1.urlrule=/xmlpserver/*
oproxy.ias1.urlrule=/xmlpserver
oproxy.ias1.urlrule=/biooffice/*
oproxy.ias1.urlrule=/biooffice
```

When you complete this Step, there are two files (`oracle_proxy.dll` and `proxy.conf`) in the folder that you created in Step 2.

4. Define the OracleAS Proxy Plug-in Registry as follows:
 - a. Edit the registry to create a registry key named `HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\IIS Proxy Adapter`.
 - b. Specify the exact location of the configuration file with the name `server_defs`, and a value pointing to the location of the configuration file, for example: `c:\proxy\proxy.conf`.
 - c. (Optional) Specify a `log_file` and `log_level`: Add a string value with the name `log_file`, and the desired location of the log file, for example, `c:\proxy\plugin.log`. Add a string value with the name `log_level`, and a value for the desired log level. Valid values are "debug", "inform", "error", and "emerg".
5. Create the "oproxy" virtual directory in IIS as follows:
 - a. Using the IIS management console, add a new virtual directory to the IIS Web site with the same physical path as that of `oracle_proxy.dll`. Name the directory "oproxy" and give it execute access.
 - b. Using the IIS management console, add `oracle_proxy.dll` as a filter in the IIS Web site. The name of the filter should be "oproxy" and its executable must point to the directory that contains `oracle_proxy.dll`, for example, `c:\proxy\oracle_proxy.dll`.

- c. Restart IIS (stop and then start the IIS server), ensuring that the filter is marked with a green arrow pointing up.
6. Check the following configuration files to remove the port 9704 because now IIS is routing all the calls to OC4J.

`\oracleBIData\web\Config\instanceconfig.xml`

`\oracleBI\xmlp\Admin\configuration\xmlp-server-config.xml`

`\OracleBI\oc4j_bi\j2ee\home\applications\biooffice\biooffice\WEB-INF\biooffice.xml`

7. To access BI Servlets from the IIS / OracleAS Proxy Plug-in, you must specify the complete URL for example:

`http://localhost/xmlpserver/login.jsp`

or

`http://localhost/biooffice/about.jsp`

and

`http://localhost/analytics/saw.dll?Answers.`

General New Features in Release 10.1.3.3.3

Oracle BI EE Release 10.1.3.3.3 introduces no new features.

General New Features in Release 10.1.3.4

Oracle BI EE Release 10.1.3.4 introduces several new features. This chapter describes those features in the following sections:

- [Section 12.1, "Overview of New Features"](#)
- [Section 12.2, "Migration of Oracle BI Discoverer Metadata"](#)
- [Section 12.3, "New Line Chart for Time Series Data"](#)
- [Section 12.4, "New Sample Oracle BI Application and Data Set"](#)
- [Section 12.5, "Integration with Oracle Enterprise Performance Management Workspace, Fusion Edition"](#)
- [Section 12.6, "Integration with Oracle Smart Space, Fusion Edition"](#)
- [Section 12.7, "Certification of Oracle WebLogic Server 10.x"](#)

12.1 Overview of New Features

Oracle BI EE Release 10.1.3.4 introduces the following new features:

- The ability to migrate relational metadata from Oracle BI Discoverer to Oracle BI Enterprise Edition Plus metadata.

For complete information, see [Section 12.2, "Migration of Oracle BI Discoverer Metadata."](#)

- A new type of line chart to plot time series data. For complete information, see [Section 12.3, "New Line Chart for Time Series Data."](#)
- A new sample Oracle BI application and data set called "Sample Sales" has been designed to better and more fully illustrate the capabilities of Oracle BI EE Release 10.1.3.4 introduces.

For complete information, see [Section 12.4, "New Sample Oracle BI Application and Data Set."](#)

- The ability to integrate Oracle BI EE and BI Publisher with EPM Workspace. EPM Workspace is a component of Foundation Services. It is the central web interface for all users to access PM system content. EPM Workspace provides a consistent, interactive and a fully thin-client environment for users to consume all EPM content. With its ease-of-use and flexibility, EPM Workspace provides users with a "windows-on-the-Web" experience.

For complete information, see [Section 12.5, "Integration with Oracle Enterprise Performance Management Workspace, Fusion Edition."](#)

- The ability to integrate Oracle BI EE and BI Publisher with Oracle Smart Space, Fusion Edition. Oracle Smart Space includes the first gadget-based user interface designed specifically for Enterprise Performance Management and Business Intelligence. Offered as part of Oracle's EPM Shared Services, Oracle Smart Space consists of a set of configurable gadgets that run on the desktop providing continuous access to financial, operational and performance related information. In addition, Oracle Smart Space includes a development toolkit for creating additional gadgets that employs common development languages and methodologies and a secure instant messaging system for shared decision making.

For complete information, see [Section 12.6, "Integration with Oracle Smart Space, Fusion Edition."](#)

- The ability to use Oracle BI EE and Oracle BI Publisher with Oracle WebLogic Server 10.x.

For complete information, see [Section 12.7, "Certification of Oracle WebLogic Server 10.x."](#)

12.2 Migration of Oracle BI Discoverer Metadata

Release 10.1.3.4 includes a utility to accelerate the migration of Oracle BI Discoverer metadata (in the form of an EEX file) to Oracle BI Enterprise Edition Plus metadata (in the form of an RPD file). The utility is a command line executable file that can be found in the `\OracleBI\Server\bin` directory. An accompanying instruction document can be found in the `\OracleBI\Server\Document` directory. This release migrates only Discoverer metadata and only for relational data sources.

12.3 New Line Chart for Time Series Data

Release 10.1.3.4 introduces a new type of line chart to plot time series data when the time intervals between data points are irregular. The time-aware line chart automatically scales the X-axis based on the time that has elapsed between data points.

Much of the information in the section "Showing Results in Charts Using Oracle BI Chart View" in *Oracle Business Intelligence Answers, Delivers, and Interactive Dashboards User Guide* applies to this new chart type, with the following exceptions and additions.

- The time-aware line chart requires a single date or date-time data column to be selected on the X-axis. The chart has a single Y-axis, but supports multiple data series. This chart type does not support drill-down on the time dimension where data types other than date or date-time are used.
- There are several formatting restrictions for this chart type. In particular, no data formatting options are available for the time axis labels or data labels. You can modify font characteristics, but any other formatting selections are ignored for this chart type. The "Stagger labels" and "Skip labels" formatting options are not supported. You cannot set start and end value or tick increments on the X-axis.
- Labels that display on the time axis are locale-sensitive. For example, month and day names use the appropriate locale-sensitive short names. Data labels for this chart type display the data value and the X-axis time value.

12.4 New Sample Oracle BI Application and Data Set

A new sample Oracle BI application and data set called "Sample Sales" has been designed to better and more fully illustrate the capabilities of Oracle BI EE Release 10.1.3.4. This new set offers an extended coverage of sample materials (reports and

RPD constructs), providing help by example in using Oracle BI EE features with specific functional cases. The content is designed to demonstrate as much as possible of Oracle BI EE capabilities and is primarily meant to help customers who want examples and templates to extract ideas from. The new sample set offers examples on two layers:

- [Section 12.4.1, "OBI EE Web Dashboard Samples"](#)
- [Section 12.4.2, "Oracle BI EE RPD Constructs Samples"](#)

This new set is meant to replace the former Paint sample content. Customers who want to retain their pre-10.1.3.4 sample Oracle BI application and any customizations that it contains should back up the BI Presentation Services Catalog, RPD file, and data files before upgrading to Oracle BI EE Release 10.1.3.4.

Note: Oracle Business Intelligence for Microsoft Office relies on Oracle Business Intelligence Web Services to fetch data for views that a user inserts in Office documents. Because this functionality returns data based on the request definition and not the view definition, it is possible when inserting views copied from a Dashboard page that the data might not match. This is a known limitation that has no workaround.

12.4.1 OBI EE Web Dashboard Samples

The new Sample Sales application includes numerous reports organized in 20 dashboard pages around the following:

- Rankings and Toppers
- Trends and History
- Tiering and Distribution
- Detailed Reports
- Lists and Segmentation (Marketing)

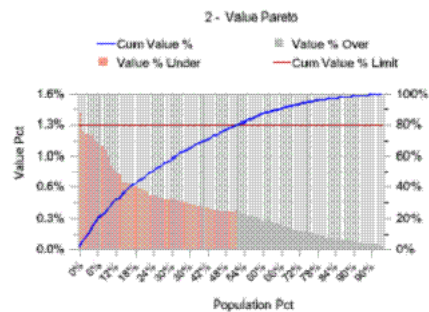
These dashboards and reports showcase specific Answers features (Answers formula building, Presentation Variables, Filtering, Layouts, Navigations, and so on) to deliver pertinent and visual insight with reduced impact on the Oracle BI EE Repository and underlying physical data model.

[Figure 12–1](#) shows a sample dashboard.

Figure 12–1 Sample Dashboard

80% of total report value is accounted for by Top 124 records, is upper 51% Pop tier of report population

Tier	Tier (in %)	1-Q1 Revenue (Sum All)	Cum Value %
Top 124 records	upper 51% Pop tier	19,983,602	80%
Remaining 113 records	lower 49% Pop tier	5,039,442	100%
Grand Total		24,903,044	100%

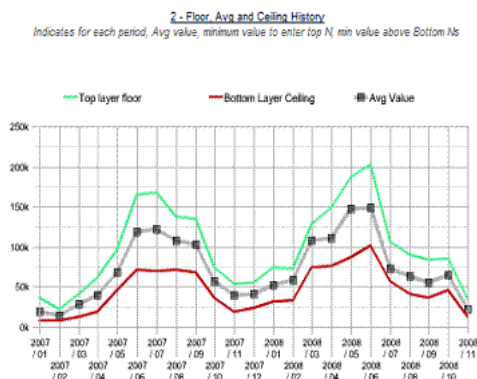


12.4.2 Oracle BI EE RPD Constructs Samples

The new Sample Sales application also includes various examples of metadata constructs, to help you design mappings for typical functional requirements when building metadata and to demonstrate best practices with this release, as described in the following list:

- Numerous logical aggregations
- Time series and Rolling X Months
- Variations, Time Span Variations, and Compounded Variations
- Run Rates and Percent of Periods
- Grain Fragmentation
- Multi Physical Sourcing
- Dimensional Snowflaking
- Canonical Time Construct (Multi LTS Facts tables)
- Data Security, Projects, and Segmentation Metadata

Figure 12–2 shows a sample graph.

Figure 12-2 Sample Graph

12.5 Integration with Oracle Enterprise Performance Management Workspace, Fusion Edition

This section contains the following topics:

- [Section 12.5.1, "Overview of the Integration"](#)
- [Section 12.5.2, "Documentation for the Integration"](#)
- [Section 12.5.3, "Installing and Configuring Oracle BI Presentation Services with EPM Workspace"](#)
- [Section 12.5.4, "Using Oracle BI Presentation Services Components in EPM Workspace, Fusion Edition"](#)

12.5.1 Overview of the Integration

In Release 10.1.3.4, the following Oracle BI EE applications have been integrated into EPM Workspace, Fusion Edition: Oracle BI Answers, Interactive Dashboards, and Delivers. In this chapter, these applications are referred to collectively as "Oracle BI Presentation Services." See the sections of this chapter for information on configuring and using this integration.

Oracle BI Publisher has also been integrated with EPM Workspace, Fusion Edition.

12.5.2 Documentation for the Integration

The following sections provide assistance in locating documentation for the integration of various Oracle BI EE applications with Oracle Enterprise Performance Management Workspace Fusion Edition:

- [Section 12.5.2.1, "Installation Documentation for the Integration"](#)
- [Section 12.5.2.2, "Integration of Oracle BI Answers, Interactive Dashboards, and Delivers, with EPM Workspace"](#)
- [Section 12.5.2.3, "Integration of Oracle BI Publisher with EPM Workspace"](#)

12.5.2.1 Installation Documentation for the Integration

For information on installing the various components, see the following guides:

- *Oracle Business Intelligence Enterprise Edition Infrastructure Installation and Configuration Guide*, for installing Oracle BI EE applications, including BI Publisher.
- *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*, for installing EPM Workspace.

12.5.2.2 Integration of Oracle BI Answers, Interactive Dashboards, and Delivers, with EPM Workspace

All the subsections in [Section 12.5](#) of this guide provide information on installing, configuring, and using Oracle BI EE components with EPM Workspace. [Table 12–1](#) describes the EPM Workspace documentation pieces.

Table 12–1 EPM Workspace Documentation Pieces

Piece	Description
<i>Reporting and Analysis New Features Guide</i>	Provides high-level information on the integration of EPM Workspace with Oracle BI EE components.
<i>EPM Workspace Administrators' Guide</i>	Provides high-level information on the Oracle BI Presentation Services Administration screen, where you perform various administration tasks for the Oracle BI EE components.
<i>EPM Workspace User's Guide</i>	Provides information on using the Oracle BI EE components from within EPM Workspace.

12.5.2.3 Integration of Oracle BI Publisher with EPM Workspace

For the integration of Oracle BI Publisher with EPM Workspace, see the following tables:

- [Table 12–2](#) for the Oracle BI Publisher documentation pieces.
- [Table 12–1](#) for the EPM Workspace documentation pieces.

Table 12–2 Oracle BI Publisher Documentation Pieces

Piece	Description
<i>Oracle Business Intelligence Publisher Administration and Developer's Guide</i>	Provides information on configuring Oracle BI Publisher for the integration with EPM Workspace.
<i>Oracle Business Intelligence Publisher Report Designer's Guide</i>	Provides information on using Oracle BI Publisher reports from within EPM Workspace and Oracle Smart Space.

12.5.3 Installing and Configuring Oracle BI Presentation Services with EPM Workspace

To install and configure Oracle BI Presentation Services for use within EPM Workspace, perform the following steps:

1. ["Registering the Oracle BI Presentation Services Components with the Hyperion Registry"](#)
2. ["Specifying the External Login for Single Sign-On with EPM Workspace"](#)
3. ["Configuring Javahost to Enable the Hyperion Common Security System Authenticator"](#)
4. ["Specifying the Hyperion_Home Environment Variable"](#)
5. ["Configuring the Repository to Use a Custom Authenticator"](#)
6. ["Configuring the Web Server for EPM Workspace"](#)

7. ["Verifying the Web Server Configuration"](#)
8. ["Provisioning Hyperion Shared Services Users in Oracle BI Presentation Services"](#)

Before configuring, install in any order Oracle BI EE, Hyperion Shared Services, and EPM Workspace, Fusion Edition. You can install the components into any appropriate installation directory. See *Oracle Business Intelligence Enterprise Edition Infrastructure Installation and Configuration Guide* and *EPM System Installation and Configuration Guide*.

12.5.3.1 Registering the Oracle BI Presentation Services Components with the Hyperion Registry

Before you can use Oracle BI Presentation Services within EPM Workspace, you must register the components. The process of registering notifies EPM Workspace that the Oracle BI Presentation Services components are installed. The following steps describe the registration process:

1. Log in to Oracle Business Intelligence.
2. Display the Oracle BI Presentation Services Administration page by clicking **Settings**, then **Administration**.
3. Click the **Manage EPM Workspace Connection** link.
4. In the **Oracle BI Presentation Services Host** field, verify the value that it contains. The field is populated with the name of the host computer for the current instance of Oracle Business Intelligence. You can specify the host name of another Oracle Business Intelligence instance to integrate with EPM Workspace.
5. In the **Port** field, enter the non-Secure Sockets Layer port number for this instance of Oracle Business Intelligence. You can change the default port number that is provided.
6. In the **SSL Port** field, enter the Secure Sockets Layer port number for Oracle Business Intelligence. You can change the default port number that is provided. Click the **Enable SSL** box beside the SSL port number to indicate that the Secure Sockets Layer port is used, rather than the non-Secure Sockets Layer port.
7. In the **Context Path** field, enter the virtual path that follows the host name in the URL for Oracle Business Intelligence.

For example, if the URL is the following:

```
http://myserver.com/analytics
```

then the Context Path is `analytics`.

8. In the **Application Server Type** box, select the type of application server that Oracle Business Intelligence is using. If the appropriate server is not in the list, then select **Other Application Server** and enter its name in the text field that is provided.
9. In the **Description** field, enter any description that you want for this connection. This description is for your information only and is not displayed elsewhere in Oracle Business Intelligence or in EPM Workspace.
10. In the **Registry Properties File** box, select one of the following values to indicate which Registry Properties file to use:
 - a. **Installed Hyperion Home:** If the computer on which this instance of Oracle Business Intelligence is installed also contains an EPM Workspace installation, then the `HYPERION_HOME` environment variable has been defined. The Registry Properties file for that Hyperion instance is used. This is the default setting.

- b. **Generate File:** When you select to generate the file, the following fields are displayed:

JDBC URL: Enter the URL to connect to the Hyperion EPM System Registry database. If you are copying the JDBC URL from an existing Registry Properties file, then ensure that you remove the "\" character in the string. The "\" character is used to escape certain special characters.

JDBC Driver: Enter the driver to connect to the Hyperion EPM System Registry database.

Username: Enter the user name to connect to the Hyperion EPM System Registry database.

Password: Enter the password to connect to the Hyperion EPM System Registry database.

For more information on these parameters, see *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

- c. **Copy File:** When you select to copy an existing Registry Properties file, the **Source File Path** field is displayed. In that field, enter the absolute path of the file to copy.

The Registry Properties file is named `reg.properties`. If you copy or generate it, then it is located in the following directory:

`OracleBIData/web/hyperion/common/config/9.5.0.0`

11. At any time in the process before you click **Register**, you can click **Finished** to cancel the registration process and return to the previous page.

When you have completed all the fields to your satisfaction, click **Register** to begin the registration process, which might take several minutes. You see status messages and error messages displayed above the **Register** button as appropriate.

12. After completing the registration process, log out of both Oracle Business Intelligence and EPM Workspace and back in to ensure that all changes have taken effect.

12.5.3.2 Specifying the External Login for Single Sign-On with EPM Workspace

An external token (or ticket) can be passed to Oracle BI Presentation Services by another system, which facilitates the use of Single Sign-on with EPM Workspace and Oracle BI. To configure for a token to be passed from EPM Workspace to Oracle BI Presentation Services for Single Sign-on, you must add or modify the `<ExternalLogon>` element of the `instanceconfig.xml` file as follows:

```
<Auth>
  <ExternalLogon enabled="true" logonPageAllowed="true">
    <ParamList>
      <Param name="UID" source="url" nameInSource="sso_token"/>
      <Param name="PWD" source="constant" value="obips.hss.sstoken"/>
    </ParamList>
  </ExternalLogon>
</Auth>
```

12.5.3.3 Configuring Javahost to Enable the Hyperion Common Security System Authenticator

You must configure Javahost for authentication, as described in the following steps:

1. On the computer on which Oracle BI Presentation Services is installed, open the `config.xml` file for editing from the following directory:
`OracleBI\web\javahost\config`
2. In the `config.xml` file, uncomment the `<OBISAuthenticatorProxy>` element.
3. Modify the values for the following elements:
 - `cssURL`: Specify the URL of the Common Security System (CSS) configuration file.
 - `useCSSURL`: Specify "yes" to use the `cssURL` parameter to load the CSS configuration file. Otherwise, the `reg.properties` file for the Hyperion Registry is used to read the CSS file from the registry.
 - `logFolder`: Specify the directory that holds CSS log files.
 - `cssDebug`: Optionally specify "yes" to enable debugging information for Common Security System authentication.
 - `processAdministrator`: Optionally specify "yes" to allow the Administrator user in Hyperion Shared Services to be authenticated in Oracle BI. The recommended value is "no".
 - `ssoTokenPassword`: Optionally specify a password to indicate that the user name that is passed to the custom authenticator is a token for Single Sign-on, as described in the previous procedure. If specified, this value must match the value that is specified as the "value" for the "PWD" parameter for the `<ExternalLogon>` element in the `instanceconfig.xml` file.
4. Save your changes and close the editor.

12.5.3.4 Specifying the Hyperion_Home Environment Variable

If you set the `useCSSURL` attribute to "no" when configuring Javahost, then specify the `HYPERION_HOME` environment variable for Javahost. This environment variable is used by the Java Virtual Machine of Javahost.

Specify the environment variable as described in these steps:

1. On the computer on which Javahost is installed, create the `hss\common\config\9.5.0.0` directory in the Javahost configuration directory.
2. Copy the `reg.properties` file to the new directory.
3. Create the `hss\common\config\9.5.0.0\resources\registry` directory in the Javahost configuration directory.
4. Copy the `RegistryLogger.properties` file from the EPM Workspace installation to the new directory.
5. Modify the `HYPERION_HOME` environment variable to have a value of "`<Javahost_Configuration_Directory>\hss`"
 - On a Windows system, set this variable by adding XML code such as the following to the `instanceconfig.xml` file:

```
<JavaHost>
  <JVMOptions>-DHYPERION_
HOME=C:/OracleBI/web/javahost/config/hss</JVMOptions>
</JavaHost>
```

- On a UNIX system, set the variable using the OracleBI/web/javahost/bin/run.sh script, as shown in the following example:

```
JAVAOPTIONS="-Xms128M -Xmx256M -Djava.awt.headless=true -
Djava.util.logging.config.file=${SawJavaHostDir}/config/logconfig.txt -
DHYPERION_HOME=${SawJavaHostDir}/config/hss"
```

12.5.3.5 Configuring the Repository to Use a Custom Authenticator

You must configure the appropriate Oracle BI EE repository file for use with a custom authenticator, as described in the following steps:

1. In the Oracle BI Administrator Tool, open the appropriate RPD file and complete the following steps to add a custom authenticator to it:

- a. From the Manage menu, select **Security**.
- b. In the Security Manager, right-click and select **New Custom Authenticator** from the popup menu.
- c. In the Custom Authenticator dialog, use the **Browse** button to specify the following value for the **Authenticator plug-in file**:

```
sawobisauthenticatorproxy3r.dll
```

This file exists in the OracleBI\web\bin directory.

- d. Also in the dialog, set **Name** to a descriptive value for this authenticator and accept the default values for **Cache persistence time** and **Number of cache entries**. You can adjust the values later as appropriate for your installation.
2. Complete the following steps to configure the authentication initialization blocks for this authenticator:
 - a. From the Manage menu, select **Variables**.
 - b. In the Variable Manager, right-click and select **New Initialization Block** from the popup menu.
 - c. In the Session Variable Initialization Block dialog, name the block "Hyperion".
 - d. In the Session Variable Initialization Block dialog, click **Edit Data Source** and select the custom authenticator that you created in Step 1.
 - e. In the Session Variable Initialization Block Data Source dialog, click **Edit Data Target**.
 - f. In the Session Variable Initialization Block Variable Target dialog, create three new session variables with the following names, all in uppercase letters: USER, GROUP, and DISPLAYNAME.
 - g. Click **OK** until all dialogs are closed.
 - h. Save your changes to the repository file.
 3. If Oracle BI Server and Javahost run on the same computer, then you have finished configuring the repository. If they run on different computers, then perform the following steps:
 - a. On the computer on which Oracle BI Server is running, open the following file for editing, or create it:

```
OracleBIData\web\config\instanceconfig.xml
```


- b. Add the following elements to the file:

```
<WebConfig>
  <ServerInstance>
    <JavaHostProxy address="host-name" />
  </ServerInstance>
</WebConfig>
```

- c. Save your changes and close the file.

12.5.3.6 Configuring the Web Server for EPM Workspace

The final step in the process is to configure the Web Server that EPM Workspace is using. You must modify the configuration of the Web Server so that requests for Oracle BI Presentation Services components are routed through the Web Server.

You can configure the Web Server either graphically or manually, as described in the following sections.

12.5.3.6.1 Configuring Graphically You can use EPM System Configurator to configure Oracle Business Intelligence components to work with the Web Server for EPM Workspace only if EPM Workspace and Oracle BI Presentation Services are deployed on a Web application server of the same type (for example, when both EPM Workspace and Oracle BI Presentation Services are deployed on Oracle Application Server).

To perform this modification graphically:

1. Launch the EPM System Configurator using one of the following methods:
 - From the Start menu in Microsoft Windows, select Programs, Oracle EPMSys, Foundation Services, and EPM System Configurator.
 - In Windows Explorer, double-click the `configtool.bat` file in the following directory:


```
HYPERION_HOME/common/config/version-number
```
 - At a Windows command prompt, change to the `HYPERION_HOME/common/config/version-number` directory and enter the following command:


```
startconfigtool.bat -console
```
2. In the Select Products to Configure dialog, select Foundation, Workspace, and Web Server Configuration, and click **Next**.
3. On the Workspace Web Server plug-in pane, an entry for Oracle BI Presentation Services is displayed, which shows its host, port, and context root. Check the box for Oracle BI Presentation Services, and check the boxes for all other products that must be configured with the Web Server, whether previously configured or not, because the configuration files are newly created and are not appended to. After checking the appropriate boxes, click **Next**.
4. Click **Next** through the remaining dialogs, and click **Finish**.

For information on using the EPM System Configurator, see the *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*.

The EPM System Configurator might suggest using different port numbers and port types for the Oracle BI Presentation Services Web Application Server. For example, if you are using the Apache HTTP Server as the Web Server and Tomcat as the Web Application Server, then the EPM System Configurator adds configuration directives to the Web Server configuration files. These directives cause Apache to use the Apache

JServe Protocol (AJP) to connect the Web Server to the Web Application Server. The AJP port should be set to the port number of the Web Server for BI Presentation Services, plus two. For example, if the BI Presentation Services Web Application is using port 9704, AJP should be enabled on port 9706. This allows the EPM System Configurator to generate automatically the correct configuration entries for the Web Server. This step is not necessary for WebLogic and WebSphere Application Servers, because they do not use AJP. For more information on enabling AJP, see *Oracle Business Intelligence Enterprise Edition Deployment Guide* or the documentation for the Web Application Server.

12.5.3.6.2 Configuring Manually If BI Presentation Services is deployed on a different type of Web application server than the type of Web application server running EPM Workspace, back up the web server's configuration files before running the EPM System Configurator to configure Oracle Business Intelligence components to work with EPM Workspace. After running the EPM System Configurator, manually restore the web server's configuration files, then add the /analytics context to the web server's configuration files. For the web server bundled with the EPM System, the configuration directives in the next paragraph add the /analytics context. For information on configuring other web servers, see *Oracle Hyperion Enterprise Performance Management System Manual Deployment Guide*.

Using a text editor, open the Web server's configuration file (httpd.conf), and add the following directives, replacing <BIEE_HOST> and <BIEE_PORT> with real values for the host name and BIEE web server port:

```
#Change modules/mod_proxy.so as needed; such as libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
    LoadModule proxy_module modules/mod_proxy.so
</IfModule>
<IfModule !mod_proxy_connect.c>
    LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
    LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
ProxyPass /analytics http://<BIEE_HOST>:< BIEE_PORT>/analytics
ProxyPassReverse /analytics http://< BIEE_HOST>:< BIEE_PORT>/analytics
```

12.5.3.7 Verifying the Web Server Configuration

Restart the Web Server and the Web Application Server that is running BI Presentation Services. Enter a URL in the following form into a new browser window:

`http://<webserver_host_name>:<webserver_port_number>/analytics`

to display the login page for BI Presentation Services. If you do not see the login page, then the Web Server configuration is likely incorrect. Return to the previous step and make the appropriate corrections.

12.5.3.8 Provisioning Hyperion Shared Services Users in Oracle BI Presentation Services

Administrators can create groups in Oracle BI Presentation Services and assign users to these groups. Such groups can be granted access to various components, such as Interactive Dashboards.

To provision Hyperion Shared Services users in these groups, create corresponding groups in Hyperion Shared Services. For example, if Oracle BI Presentation Services has a group called "Sales", create the same group "Sales" in Hyperion Shared Services

and add the users to this group. The Hyperion Shared Services users now belong to the "Sales" group in Oracle BI Presentation Services.

Similarly for provisioning users as Administrators in Oracle BI Presentation Services, create the Presentation Services Administrators group in Hyperion Shared Services and add Hyperion Shared Services users to this group.

12.5.4 Using Oracle BI Presentation Services Components in EPM Workspace, Fusion Edition

After you configure Oracle BI Presentation Services to run within EPM Workspace, you can begin using the components there, as described in the following sections:

- [Section 12.5.4.1, "Specifying Oracle BI Preferences Within EPM Workspace"](#)
- [Section 12.5.4.2, "Performing Oracle BI Presentation Services Administration Tasks Within EPM Workspace"](#)
- [Section 12.5.4.3, "Launching Oracle BI Presentation Services Components in EPM Workspace"](#)
- [Section 12.5.4.4, "Using BI Answers"](#)
- [Section 12.5.4.5, "Using BI Interactive Dashboards"](#)
- [Section 12.5.4.6, "Using BI Delivers"](#)

12.5.4.1 Specifying Oracle BI Preferences Within EPM Workspace

As you work with the Oracle BI Presentation Services components in EPM Workspace, you likely want to specify preferences to customize your experience.

To specify preferences, complete the following steps:

1. From the File menu, select **Preferences**.
2. From the left-hand list, select **Oracle BI**.
3. On the My Account page, update the information as appropriate.

The Locale and User Interface Languages lists are read-only, because you cannot modify them. The locale and language are inherited from EPM Workspace. See *Oracle Enterprise Performance Management Workspace, Fusion Edition Administrator's Guide* for information on setting the locale and language.

4. When you are finished, click **OK**.

12.5.4.2 Performing Oracle BI Presentation Services Administration Tasks Within EPM Workspace

To allow users to work effectively in EPM Workspace, you might need to perform Oracle BI Presentation Services Administration tasks.

To perform these tasks, complete the following steps:

1. From the Navigate menu, select **Administer**, then **BI Presentation Services**.
2. On the Oracle BI Presentation Services Administration page, perform the appropriate activities, such as managing the Presentation Catalog and interactive dashboards.

12.5.4.3 Launching Oracle BI Presentation Services Components in EPM Workspace

To launch Oracle BI Presentation Services Components, use the Navigate menu in the upper-left corner of the EPM Workspace window.

Select **Navigate**, then **Applications**, then the appropriate component: Oracle BI Answers, Oracle BI Interactive Dashboards, or Oracle BI Delivers.

12.5.4.4 Using BI Answers

BI Answers runs in a new tab in the EPM Workspace and adds various menu items for creating and opening content, as described in the following list:

1. To launch from the Navigate menu, select **Applications**, then **Oracle BI Answers**.
2. From the File menu, select **New**, then the appropriate BI Answers item such as **Request**, **Filter**, or **Dashboard Prompt**.

Tip: To create a new database request, select **New** from the File menu, then select **Request**. Select **Create Direct Request** in the resulting dialog.

3. From the File menu, select **Open**, then **Oracle BI Catalog** to display a dialog from which you can open a BI Answers item.
4. From the File menu, select **Save** to begin the process of saving a BI Answers item. You can also use the Save icon on the toolbar.
5. From the Tools menu, select **Manage Catalog** to edit and organize the content in the Presentation Catalog.
6. From the Tools menu, select **Install**, then **Oracle BI for Microsoft Office** to begin the process of installing that component.

12.5.4.5 Using BI Interactive Dashboards

Oracle BI Interactive Dashboards runs in a new tab in EPM Workspace and adds various menu items for navigating in and editing dashboards, as described in the following list:

1. To launch from the Navigate menu, select **Applications**, then **Oracle BI Interactive Dashboards**.
2. From the Dashboards menu, select the items for navigating among dashboards.
3. From the Edit menu, select **Edit Dashboard** to display the Dashboard Editor, or use the other options on the Edit menu to perform appropriate Interactive Dashboards functions. These options are the same as those found on the Page Options menu that is available when Interactive Dashboards runs as part of Oracle Business Intelligence Enterprise Edition.
4. To see an alert (if one is available), select **Alerts** from the View menu or click the Alerts icon on the toolbar.

12.5.4.6 Using BI Delivers

BI Delivers runs in a new tab in EPM Workspace and adds various menu items for creating and opening iBots, as described in the following list:

1. To launch from the Navigate menu, select **Applications**, then **Oracle BI Delivers**.
2. From the File menu, select **New**, then **iBot**.

3. From the File menu, select **Save** to begin the process of saving the iBot.
4. From the File menu, select **iBots Acting on My Behalf** to see any iBots that are available.
5. To see an alert (if one is available), select **Alerts** from the View menu or click the Alerts icon on the toolbar.

12.6 Integration with Oracle Smart Space, Fusion Edition

This section contains the following topics:

- [Section 12.6.1, "Overview of the Integration"](#)
- [Section 12.6.2, "Documentation for the Integration"](#)
- [Section 12.6.3, "Performing Oracle BI Tasks in Smart Space"](#)
- [Section 12.6.4, "Downloading Oracle Smart Space and Smart View from Oracle BI Presentation Services"](#)

12.6.1 Overview of the Integration

Release 10.1.3.4 introduces the following two types of integration of Oracle BI EE with Oracle Smart Space:

- [Section 12.6.3, "Performing Oracle BI Tasks in Smart Space"](#): You can launch and interact with Oracle BI Interactive Dashboards and Answers requests in the Smart Space Content Viewer and Smart Book gadget and add them to the Favorites gadget. You can use Smart Space Collaborator to share Oracle BI content with other Smart Space users. You can also configure iBot results to be sent as Smart Space notifications and view these in the Notification gadget.
- [Section 12.6.4, "Downloading Oracle Smart Space and Smart View from Oracle BI Presentation Services"](#): You can set up for users to download Oracle Smart Space and Oracle Smart View for Microsoft Office from within Oracle BI.

12.6.2 Documentation for the Integration

The following sections provide assistance in locating documentation for the integration of various Oracle BI EE applications with Oracle Smart Space:

- [Section 12.6.2.1, "Installation Documentation for the Integration"](#)
- [Section 12.6.2.2, "Integration of Oracle BI Answers, Interactive Dashboards, and Delivers with Oracle Smart Space"](#)
- [Section 12.6.2.3, "Integration of Oracle BI Publisher with Oracle Smart Space"](#)

12.6.2.1 Installation Documentation for the Integration

For information on installing the various components, see the following books:

- *Oracle Business Intelligence Enterprise Edition Infrastructure Installation and Configuration Guide*, for installing Oracle BI EE applications, including BI Publisher.
- *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*, for installing Oracle Smart Space.

See *Oracle Smart Space Configuration Guide* for additional information on installing and configuring Oracle Smart Space with Oracle BI EE components, including BI Publisher

12.6.2.2 Integration of Oracle BI Answers, Interactive Dashboards, and Delivers with Oracle Smart Space

The following tables describes where you can find information for the integration of Oracle BI Answers, Oracle BI Interactive Dashboards, and Oracle BI Delivers with Oracle Smart Space.

[Table 12–3](#) describes the Oracle BI EE documentation pieces.

Table 12–3 Oracle BI EE Documentation Pieces

Piece	Description
Section 12.6, "Integration with Oracle Smart Space, Fusion Edition" in this guide	Provides information on using Oracle BI EE components within Oracle Smart Space and on downloading Oracle Smart Space and Smart View from Oracle BI Answers.
Tech Note 559596.1	Using Oracle Delivers iBots, you can send notifications to Smart Space users. To set up iBots in Oracle BI Delivers for use with Smart Space, see Tech Note 559596.1 on My Oracle Support. Oracle BI Delivers uses intelligence bots called iBots which are software-based agents that detect specific results. iBot messages can be sent as notifications to Smart Space Collaborator users

[Table 12–4](#) describes the Oracle Smart Space documentation pieces.

Table 12–4 Oracle Smart Space Documentation Pieces

Piece	Description
<i>Oracle Smart Space New Features Guide</i>	Provides high-level information on the integration of Oracle Smart Space with Oracle BI EE components, including BI Publisher.
<i>Oracle Smart Space Configuration Guide</i>	Provides information on installing and configuring Oracle Smart Space with Oracle BI EE components, including BI Publisher.
<i>Oracle Smart Space Administrator's Guide</i>	Provides information on configuring services to access Oracle BI EE content, including BI Publisher reports, from Oracle Smart Space.
<i>Oracle Smart Space User's Guide</i>	Provides information on accessing Oracle BI EE content, including BI Publisher reports, from Oracle Smart Space.

12.6.2.3 Integration of Oracle BI Publisher with Oracle Smart Space

For the integration of Oracle BI Publisher with Oracle Smart Space, see the following tables:

- [Table 12–2](#) for the Oracle BI Publisher documentation pieces.
- [Table 12–4](#) for the Oracle Smart Space documentation pieces.

12.6.3 Performing Oracle BI Tasks in Smart Space

Once Smart Space is connected to Oracle BI, you can use Windows Explorer to browse the Oracle BI Presentation Catalog and see the folders (Shared Folders and My

Folders) and documents (Dashboard and Answers request) that are available to you. The following sections provide information on the tasks that you can perform for Oracle BI from within Smart Space:

- [Section 12.6.3.1, "Tasks from Right-Click Menus"](#)
- [Section 12.6.3.2, "Folder Management"](#)
- [Section 12.6.3.3, "Oracle BI Delivers iBots"](#)
- [Section 12.6.3.4, "Authentication"](#)

12.6.3.1 Tasks from Right-Click Menus

From Windows Explorer, right-click a document to see the available Oracle BI functions.

The following list details the right-click menus available for Oracle BI documents in Smart Space.

- Answers Requests:
 - Open: Opens the Answers request in the Smart Space Content Viewer for viewing and interacting. The Oracle BI masthead is not available.
 - Open in Browser: Opens the Answers request in Oracle BI in a browser and loads the request for viewing and interacting. The Oracle BI masthead is not available.
 - Edit: Opens Oracle BI in a browser and loads the Answers request for editing. The Oracle BI masthead is available for navigating to other Oracle BI pages.
 - Cut: Places the Answers request on the Windows Clipboard for moving into another folder within the Presentation Catalog on the same Oracle BI Server.
 - Copy: Places a copy of the Answers request on the Windows Clipboard for copying into another folder within the Presentation Catalog on the same Oracle BI Server.
 - Paste: Pastes the Answers request into a folder, after the request has been cut or copied to the Windows Clipboard.
 - Delete: Removes the Answers request from the Presentation Catalog.
 - Rename: Renames the Answers request within the Presentation Catalog. When Answers requests are renamed, Smart Space automatically preserves references to the previous name of this item.

Access to cut, copy, paste, delete and rename is based on the privileges that are defined for you in the Oracle BI Presentation Catalog.

- Interactive Dashboards:
 - Open: Opens the dashboard in the Smart Space Content Viewer for viewing and interacting. The Oracle BI masthead is not available.
 - Edit: Opens the Dashboard Editor screen of Oracle BI in a browser with the dashboard loaded for editing.
 - Open in Browser: Opens the Interactive Dashboards screen in Oracle BI in a browser and loads the dashboard for viewing and interacting. The Oracle BI masthead is available for navigating to other Oracle BI pages.

My Dashboard is displayed directly underneath the "My Folders" folder.

12.6.3.2 Folder Management

Smart Space's integration with Windows Explorer also enables creation, renaming and deletion of folders (both Shared Folders and My Folders). Access to these functions is based on the privileges that are defined for you in the Oracle BI Presentation Catalog.

12.6.3.3 Oracle BI Delivers iBots

Using Oracle Delivers iBots, you can send notifications to Smart Space users. To set up iBots in Oracle BI Delivers for use with Smart Space, see Tech Note 559596.1 on My Oracle Support.

Oracle BI Delivers uses intelligence bots called iBots which are software-based agents that detect specific results. iBot messages can be sent as notifications to Smart Space Collaborator users

12.6.3.4 Authentication

Smart Space supports the following authentication methods for Oracle BI: Oracle BI native authentication and OracleAS Single Sign-On.

12.6.4 Downloading Oracle Smart Space and Smart View from Oracle BI Presentation Services

In Release 10.1.3.4, you can download one of the following Microsoft Office Add-ins from within Oracle BI Presentation Services: Oracle BI for Microsoft Office, Smart View for Microsoft Office, or Oracle Smart Space. This section contains the following topics:

- [Section 12.6.4.1, "Setting Up to Download Oracle Smart Space and Smart View from Oracle BI Presentation Services"](#)
- [Section 12.6.4.2, "Initiating the Download"](#)

For more information, refer to the following:

- For information on Oracle Smart Space, see the *Oracle Smart Space User's Guide*.
- For information on Oracle Hyperion Smart View for Office, Fusion Edition, see the *Oracle Hyperion Smart View for Office User's Guide*.
- For information on downloading Oracle BI for Microsoft Office, see [Chapter 31, "Installing Oracle BI for Microsoft Office in Release 10.1.3.3."](#)

12.6.4.1 Setting Up to Download Oracle Smart Space and Smart View from Oracle BI Presentation Services

Before you can see the menu options for downloading an add-in, you must perform configure tasks for each one.

12.6.4.1.1 Smart View Complete the following steps in the specified order to set up to download Smart View. While the order of installation is not important, Provider Services must be started before users can connect to Oracle BI EE using Smart View.

1. Install and start Provider Services. Smart View uses Provider Services for accessing Oracle BI EE content.

For information on installing Provider Services, see *Oracle Hyperion EPM System Installation Start Here Guide*.

2. Install Oracle BI.

For information, see *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

3. Set up a connection to Oracle BI EE using the Smart View Data Source Manager in the Smart View client.

For information, see the chapter on connecting to data sources in *Oracle Smart View, Fusion Edition User's Guide*.

Note: There is no separate administration tool for configuring data sources for use with Smart View. Administrators can install Smart View using the standalone installer, if they want to set up connections before enabling the Smart View download for users. The assumption is that the Oracle BI installation has been completed and the services are running along with the Provider Services.

4. Update the Oracle BI EE instanceconfig.xml file with Smart View Client download information, as shown in the following example:

```
<WebConfig>
  <ServerInstance>
    <SmartViewInstallerURL>http://<URL to download SmartView
client>/</SmartViewInstallerURL>
  </ServerInstance>
</WebConfig>
```

For information on the instanceconfig.xml file, see *Oracle Business Intelligence Presentation Services Administration Guide*.

5. Restart Oracle BI to see the new menu option for Smart View.

12.6.4.1.2 Oracle Smart Space Complete the following steps to set up to download Oracle Smart Space.

1. Install Oracle BI.

For information, see *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.

2. Install Oracle Smart Space.

For information, see *Hyperion EMP System Installation and Configuration Guide*.

3. Ask the Oracle Smart Space administrator to configure Smart Space with Oracle BI registration information.

For information, see *Oracle Smart Space, Fusion Edition Administrator's Guide*.

4. Update the instanceconfig.xml file with information to generate the link for Oracle Smart Space start.html page for installing the Smart Space client, as shown in the following example:

```
<WebConfig>
  <ServerInstance>
    <SmartSpaceInstallerURL>http://<URL to download SmartSpace
client>/</SmartSpaceInstallerURL>
  </ServerInstance>
</WebConfig>
```

5. Restart Oracle BI to see the new menu option.

12.6.4.2 Initiating the Download

To initiate the download of an add-in, perform these steps:

1. Log in to Oracle Business Intelligence.
2. Click the More Products link, then click one of the following:
 - **Download Oracle BI for Microsoft Office**
See [Chapter 31.4, "Installing the Oracle BI Add-In for Microsoft Office"](#) for information on installing the Oracle BI add-in, which is not a new feature for Release 10.1.3.4.
 - **Download Smart View for Microsoft Office**
 - **Download Smart Space**

12.7 Certification of Oracle WebLogic Server 10.x

You can use Oracle BI EE and Oracle BI Publisher with the Oracle WebLogic Server. The following sections describes the steps that are required to configure for that application server.

12.7.1 Configuring Oracle BI EE for WebLogic Server

The following steps provide information on configuring to use WebLogic Server with Oracle BI EE. See the WebLogic Server documentation for additional information.

1. Create a new WebLogic Server domain using a JDK such as Sun JDK 1.5.x or higher, in place of JRockit.
2. Create an exploded archive directory from the analytics.ear or analytics.war file using the new domain, by following these steps:
 - a. Copy the analytics.ear or analytics.war file in the \OracleBI\web directory to a destination directory, as shown in the following example.

```
-mkdir c:mydestination\BIEE\  
-cp c:\OracleBI\web\analytics.war to c:mydestination\BIEE\
```

- b. Manually unpack the analytics.ear or analytics.war file using a jar command, as shown in the following example.

```
jar -xvf manual\generic\analytics.war
```

By running the jar command, you create a directory called "analytics." This is the directory where the application is deployed.

Note the following about exploded archive directories: Do not deploy the analytics.war or analytics.ear file on the WebLogic Server by uploading it from the WebLogic console, because the console deploys the application (or Web module) in an archived file format.

- -
 3. If the BI Presentation Services Server runs on a different computer from the WebLogic Server, then edit the web.xml file for the WebLogic Server to account for this difference.
 4. In a Web browser, start the WebLogic Administration Console, as shown in the following example.

```
http://hostname:7001/console
```

5. If you have not previously done so, in the Change Center of the Administration Console, click **Lock & Edit**.
6. In the left pane of the Administration Console, click **Deployments**.
7. Install the deployment as an application by completing the following steps:
 - a. In the right pane, click **Install**.
 - b. Select c:\mydestination\BIEE and click **Next**.
 - c. Select **Install this deployment as an application** and click **Next**.
 - d. Select **I will make the deployment accessible from the following location** from Source accessibility.
 - e. Click **Finish**.
8. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
9. Start the service by clicking **Deployments**, then **analytics**.

12.7.2 Configuring Oracle BI Publisher for WebLogic Server

The following steps provide information on configuring to use WebLogic Server with Oracle BI Publisher. See the WebLogic Server documentation for additional information.

1. Create a new WebLogic Server domain using a JDK such as Sun JDK 1.5.x or higher, in place of JRockit.
2. Create an exploded archive directory from the xmlpserver.ear or xmlpserver.war file using the new domain, by following these steps:
 - a. Copy the xmlpserver.ear or xmlpserver.war file in the \xmlpserver\web directory to a destination directory, as shown in the following example.

```
-mkdir c:\mydestination\xmlpserver\  
-cp c:\xmlpserver\web\xmlpserver.war to c:\mydestination\xmlpserver\
```

- b. Manually unpack the xmlpserver.ear or xmlpserver.war file using a jar command, as shown in the following example.

```
jar -xvf manual\generic\xmlpserver.war
```

By running the jar command, you create a directory called "xmlpserver." This is the directory where the application is deployed.

Note the following about exploded archive directories: Do not deploy the xmlpserver.war or xmlpserver.ear file on the WebLogic Server by uploading it from the WebLogic console, because the console deploys the application (or Web module) in an archived file format. This is problematic to a BI Publisher configuration, because you must update WEB-INF/xmlp-server-config.xml manually after the deployment. To work around this issue, use an "exploded archive" directory.

3. If you need changes to the BI Publisher configuration files, such as updating the catalog path in the xmlp-server-config.xml file, then make the changes before deployment.
 4. In a Web browser, start the WebLogic Administration Console, as shown in the following example.

```
http://hostname:7001/console
```

5. If you have not previously done so, in the Change Center of the Administration Console, click **Lock & Edit**.
6. In the left pane of the Administration Console, click **Deployments**.
7. Install the deployment as an application by completing the following steps:
 - a. In the right pane, click **Install**.
 - b. Select c:\mydestination\xmlpserver and click **Next**.
 - c. Select **Install this deployment as an application** and click **Next**.
 - d. Select **I will make the deployment accessible from the following location** from Source accessibility.
 - e. Click **Finish**.
8. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
9. Configure the BI Publisher repository and install fonts as described in *Oracle Business Intelligence Publisher Installation Guide*.
10. Start the service by clicking **Deployments**, then **xmlpserver**.

General New Features in Release 10.1.3.4.1

Oracle BI EE Release 10.1.3.4.1 introduces the following new features:

- [Section 13.1, "Configuring for Connections to EPM Workspace."](#)
- [Section 13.2, "Validating the Catalog."](#)

13.1 Configuring for Connections to EPM Workspace

You can configure Oracle BI EE to work with EPM Workspace, as described in [Section 12.5, "Integration with Oracle Enterprise Performance Management Workspace, Fusion Edition."](#) In addition, you can use the `ManageEPMWorkspaceConnection` privilege to control whether the "Manage EPM Workspace Connection" link is displayed on the Administration page for certain users.

You can include the following element in the `instanceconfig.xml` file to control whether Oracle BI EE connections to Workspace are allowed:

```
<ServerInstance>
<ManageEPMWorkspaceConnection>true</ManageEPMWorkspaceConnection>
</ServerInstance>
```

13.2 Validating the Catalog

Over time, inconsistencies can develop in the catalog as links are broken, users are deleted, or NFS file system issues are encountered. These inconsistencies can eventually lead to incorrect behavior, such as the inability to edit an agent's recipient list. You can periodically take the production system offline and validate the catalog, to be informed of and to take corrective action on inconsistencies.

This section contains the following topics about validating that catalog:

- [Section 13.2.1, "Process: Validating the Catalog."](#)
- [Section 13.2.2, "Important Guidelines for Validating the Catalog."](#)
- [Section 13.2.3, "Performing a Basic Validation of the Catalog."](#)
- [Section 13.2.4, "Specifying the Elements for Validating the Catalog."](#)

13.2.1 Process: Validating the Catalog

The process of validating the catalog involves creating a report for the catalog in offline mode and seeing the objects that require adjustment or removal. You might fix some objects manually in offline mode. Then run the validate operation again to allow the system to "clean" by deleting any unnecessary objects. You repeat the process of

creating the report, manually fixing errors, and cleaning the catalog until it is validated.

13.2.2 Important Guidelines for Validating the Catalog

Before you validate the catalog, keep the following guidelines in mind:

- When you turn on any validation of the catalog, all ACLs (that is, all privileges and every item's permissions) are "scrubbed," which means that dead accounts are removed from them and any changed items are written to disk. Therefore, even if you simply create a report without fixing any broken items automatically, you might find that the catalog is still extensively "changed."
- Before validating the catalog in a clustered environment, do one of the following:
 - Shut down the Presentation Services cluster and run the validation directly against the cluster's catalog.
 - Make a copy of the cluster's catalog and run the validation against that copy.

Before using the 7-Zip utility to make a copy of a catalog, first shut down all nodes in the Presentation Services cluster or put all nodes in that cluster into Maintenance Mode (which is the recommended approach).

Be aware that any changes that are made to the catalog online concurrently to the validation process are not included in the validation.

While backing up the catalog is always good practice, there is no practical difference between running validate against the catalog directly versus running the validation on a backup copy.

13.2.3 Performing a Basic Validation of the Catalog

You can perform a basic validation of the catalog on an ad-hoc basis as needed, immediately before pushing content from a development environment to a production environment, or per a regular schedule, such as on the first Tuesday of every month.

To validate the catalog:

1. Stop Presentation Services.
2. Back up the catalog by using the 7-Zip utility to create a compressed file for it.
3. Create a backup copy of the instanceconfig.xml file.
4. Edit the instanceconfig.xml file so that it contains the appropriate elements for performing the validation. You must set the elements to perform the tasks for creating the report and "cleaning" the catalog at the appropriate times.

For information on these elements, see [Section 13.2.4, "Specifying the Elements for Validating the Catalog."](#)

5. Start Presentation Services to run the validation according to the values that you specified in the instanceconfig.xml file.
6. Edit the instanceconfig.xml file again so that it contains the appropriate elements for performing the validation. You must set the elements to perform the tasks for creating the report and "cleaning" the catalog at the appropriate times.
7. Repeat Steps 4 through 6 until the catalog is validated.
8. Stop Presentation Services.

9. Create a backup copy of the instanceconfig.xml file in which you added the validation elements and rename the file similar to instanceconfig_validate.xml. In this way, you have a version of the file to use as a starting point for subsequent validations.
10. Restore the backup version of the instanceconfig.xml that you created earlier to use as the current version.
11. Start Presentation Services.

13.2.4 Specifying the Elements for Validating the Catalog

As part of the process of validating the catalog, you include elements in the instanceconfig.xml file that run the validation when you restart Presentation Services. The following procedure describes how to edit the instanceconfig.xml file to include these elements.

To specify the element for validating the catalog:

1. Open the instanceconfig.xml file for editing.
2. Locate the Catalog section in which you must add the elements that are described in [Table 13-1](#).
3. Include the elements and their ancestor element as appropriate, as shown in the following example. In this example, the validation runs when Presentation Services starts, and Presentation Services is stopped when the validation is complete. Inconsistent accounts (such as those for deleted users), links, and objects are removed. Inconsistent users' home directory names are logged but directories are not removed.

```
<ServerInstance>
  <Catalog>
    <Validate>OnStartupAndExit</Validate>
    <ValidateAccounts>Clean</ValidateAccounts>
    <ValidateHomes>Report</ValidateHomes>
    <ValidateItems>Clean</ValidateItems>
  </Catalog>
</ServerInstance>
```

Caution: Include only one Catalog element in the instanceconfig.xml file or unexpected results might occur. Unless expressly noted, include most nodes (such as that for the Catalog element) in an XML document only once.

4. Save your changes and close the file.

Table 13–1 Elements for Validating the Catalog

Element	Description	Default Value
Validate	<p>Performs the validation of the catalog according to the values of the other Validate-related elements in this section. Values are described in the following list:</p> <ul style="list-style-type: none"> ■ None — Performs no validation. ■ OnStartupAndExit — When Presentation Services starts, performs the validation, performs Report or Clean operations, then stops Presentation Services. You run through multiple cycles of Report, Clean, Report, and so on for each element (such as ValidateAccounts, ValidateHomes, ValidateItems, and ValidateLinks) until the catalog is validated. <p>If this value is not None, then all privileges and each object's ACLs in the entire catalog are cleaned of terminated accounts, regardless of the settings of the other Validate-related elements.</p>	None
ValidateAccounts	Verifies that all information about users, roles, and groups in the catalog is consistent. Values are described in the list after this table.	None
ValidateHomes	<p>Verifies that all information about home directories in the catalog is consistent. Values are described in the list after this table.</p> <p>ValidateHomes is executed only if ValidateAccounts is set to either Report or Clean.</p>	None
ValidateItems	Verifies that all information about objects in the catalog is consistent. Values are described in the list after this table.	None

The elements have the values that are described in the following list:

- None — Specifies that no validation is performed.
- Report — Specifies that details about each inconsistent object are written to the sawlog.log file.
- Clean — Specifies that details about each inconsistent object are written to the sawlog.log file and that each object is removed from the catalog.

General New Features in Release 10.1.3.4.2

Oracle BI EE Release 10.1.3.4.2 introduces new features. This chapter describes those features in the following sections:

- [Section 14.1, "Support for SAP/BW Variables"](#)
- [Section 14.2, "Enhancements for Catalog Manager"](#)

14.1 Support for SAP/BW Variables

You can use SAP/BW variables when using SAP/BW as a data source. There are five types of SAP/BW variables — Characteristic value, Hierarchy, Hierarchy node, Text, and Formula variables. Oracle BI EE supports two of the types, Characteristic value and Hierarchy node. Cube variable objects are imported into the physical layer when metadata is imported from BEx Queries in SAP/BW data sources. Typically, you do not edit these objects, except to keep them synchronized with the BEx queries in the data source, and except to specify overrides for key characteristic values.

When setting up a variable in the Variable Editor of the BEx Query Designer, you must define the processing type for the variable and describe a setting for what the variable represents. For both Characteristic value and Hierarchy node variables, the following processing types are supported — Manual Input/Default Values, Customer Exit, Authorizations, and SAP Exit. For Characteristic value variables, the following options for the "Variable Represents" setting are supported — Single Value, Multiple Single Values, Interval, and Selection Option. For Hierarchy node variables, the following options for the "Variable Represents" setting are supported: — Single Value, and Multiple Single Values.

When the BI Server generates a physical request against a BEx query with variables, applicable filters in the logical request are translated to the appropriate MDX syntax with the SAP Variable clause. Filters can still be created within Oracle BI EE and passed down to the underlying BEx Query even if no variables have been defined. This functionality is not changed.

The following list provides additional suggestions for working with SAP/BW variables within Oracle BI EE:

- The behavior for variables is the same for both the Level and the Key.
- The value for 'MAX_MEMBER_PER_LEVEL' might need to be increased (based on the configuration) in the NQConfig.INI file.
- The restriction behavior of the variable (inclusion or exclusion) is sometimes defined in the BEX Query itself. Hence, even if the operator "is equal to / is in: is used in Oracle BI EE, then the value might be excluded from the result if in the BEx query the variable is used as exclusion filter.

- The settings used in the Default Values tab of the Variable Editor in the BEx Query Designer are not imported to Oracle BI EE.

For additional information about setting up SAP/BW variables, see your SAP/BW documentation.

14.2 Enhancements for Catalog Manager

This section describes the following enhancements for Catalog Manager:

- [Section 14.2.1, "General Enhancements"](#)
- [Section 14.2.2, "Creating Reports"](#)
- [Section 14.2.3, "Obtaining Audit Trail Information for Objects"](#)

14.2.1 General Enhancements

The following list outlines enhancements to Catalog Manager:

- Search and replace functionality is case-sensitive.
- You can search and replace table column names in addition to subject areas and tables.
- When you display a report of objects, you can edit the columns that are shown.

14.2.2 Creating Reports

This section replaces the section on "Creating Reports to Display Presentation Catalog Data Using Catalog Manager" from *Oracle Business Intelligence Presentation Services Administration Guide*.

You can create reports to display catalog data for all catalog object types. You can either display the report on the screen or save it to a file.

For example, to find out which dashboards are using an analysis, you can run a Dashboard report including analyses, and search that report for the analysis. Or, to find analyses that are affected by a changed column in an RPD table, you can run an Analysis report that includes all columns and formulas, and then search the report for the items that must then be replaced in Catalog Manager.

When you create a report, a blank or empty field is exported as a tab character. If you are creating a report with the default of a tab as the field separator, then two tab characters in the report file indicate a blank field.

To create a report that displays catalog data:

1. In Catalog Manager, open the catalog. To create a report that shows the SQL statement that is sent to the Oracle BI Server for the object, open the catalog in online mode.
2. Select the top folder for the catalog.
3. From the **Tools** menu, select **Create Report**.
4. Select the catalog object type to create a report on.
5. To eliminate any rows that are the same from the report, select the **Distinct** box.
6. Specify the columns to appear in the report in the Columns in Report list. Use the left and right-arrow buttons (< and >) to move the columns between the Available

Columns list and the Columns in Report list, and the plus and minus buttons (+ and -) to set the order in which columns are displayed in the report.

7. Click **OK**.
8. Repeat Steps 4 through 7 until the report contains the appropriate columns.
9. To save the report to a file, in the **Save report to** field, specify the path name of the file. Click the **Browse** button to display the Save As dialog for selecting the path name (if the file does not exist, then it is created).
10. Click **OK** to overwrite an existing file or click **Cancel** and enter a new name to save to a new file.
11. Select **Excel Format** to specify to create a file that can be imported into Microsoft Excel 2003 or later.
12. Click **OK**.

14.2.3 Obtaining Audit Trail Information for Objects

If you have many different users working with objects in the catalog, then you might want to see audit trails for those objects to determine what changes were made and by whom. To obtain an audit trail, you use the facility that is described in Chapter 9 "Using Oracle BI Presentation Services Logging" of *Oracle Business Intelligence Presentation Services Administration Guide*.

To create an audit trail file, you use the FilterRecord element with various attributes. Keep in mind that when you turn on the creation of audit trails, the performance of the system is likely to decrease, as resources are required to write data to disk for the creation of the audit trail files.

The following list describes various types of information that you can capture in audit trail files by including the FilterRecord element in the Filters section of the logconfig.xml file:

- **Changes to ACLs**

Use an element such as the following one to track changes to ACLs for objects. This setting tracks anyone who makes any changes to ownership or permissions of any object. For example, suppose that the report /shared/sales/MyFavReport was owned by Fred, and Steve had Read access. If an administrator changes the owner to Bill or also gives Write access to Sue, then this audit trail logs those changes.

```
<FilterRecord writerClassGroup="File" path="saw.catalog.acl.auditChanges"
information="51" warning="100" error="100" security="41" />
```

- **Changes to Object Contents**

Use an element such as the following one to track modifications to object contents. For example, if Fred changes the contents of the MyFavReport object, then this audit trail logs those changes.

```
<FilterRecord writerClassGroup="File" path="saw.catalog.local.writeObject"
information="51" warning="100" error="100" security="41" />
```

- **Logging In and Out**

Use an element such as the following one to obtain messages that relate to users logging in and out:

```
<FilterRecord writerClassGroup="File" path="saw" information="30" warning="30"
error="100" security="56"/>
```

Note that the setting of the security attribute to 56 is important for obtaining the appropriate information.

Part II

New Features for Oracle BI Publisher

Part II contains the following chapters:

- [Chapter 15, "Oracle BI Publisher Template Builder Enhancements in Release 10.1.3.3"](#)
- [Chapter 16, "Flash Templates for Oracle BI Publisher in Release 10.1.3.3"](#)
- [Chapter 17, "Oracle BI Publisher PowerPoint Output in Release 10.1.3.3"](#)
- [Chapter 18, "Oracle BI Publisher Integration with Oracle BI Discoverer in Release 10.1.3.3"](#)
- [Chapter 19, "Performance Improvements for Oracle BI Publisher in Release 10.1.3.3"](#)
- [Chapter 20, "Converting Reports from Oracle Reports to Oracle BI Publisher in Release 10.1.3.3"](#)
- [Chapter 21, "Support for Digital Signature in PDF Documents in Release 10.1.3.3"](#)
- [Chapter 22, "Support for PostScript Printers in Release 10.1.3.3"](#)
- [Chapter 23, "Additional New Features in BI Publisher Release 10.1.3.3"](#)
- [Chapter 24, "New Features in BI Publisher Release 10.1.3.3.1"](#)
- [Chapter 25, "New Features in BI Publisher Release 10.1.3.3.2"](#)
- [Chapter 26, "New Features in BI Publisher Release 10.1.3.3.3"](#)
- [Chapter 27, "BI Publisher Web Services, Releases 10.1.3.3.1, 10.1.3.3.2, and 10.1.3.3.3"](#)
- [Chapter 28, "New Features in BI Publisher Release 10.1.3.4"](#)
- [Chapter 29, "New Features in BI Publisher Release 10.1.3.4.1"](#)
- [Chapter 30, "New Features in BI Publisher Release 10.1.3.4.2"](#)

Oracle BI Publisher Template Builder Enhancements in Release 10.1.3.3

This chapter describes enhancements to the Oracle BI Publisher Template Builder that were introduced in Release 10.1.3.3 and that apply to Releases 10.1.3.3 through 10.1.3.4. Information about these features is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

There are many new and exciting enhancements to the Oracle BI Publisher Template Builder. These new features enable users to use Discoverer worksheets as data sources, generate reports in Microsoft PowerPoint format, and to use more powerful formatting facilities and charts. The new features include the following:

- Integration with Discoverer
- Support for Microsoft PowerPoint Output
- Advanced Form Field Support
- Enhanced Chart Builder Features

This chapter includes the following sections:

- [Section 15.1, "Changes to the Oracle BI Publisher Menu"](#)
- [Section 15.2, "Integration with Oracle BI Discoverer"](#)
- [Section 15.3, "Microsoft PowerPoint Output Support"](#)
- [Section 15.4, "Advanced Form Field Support"](#)
- [Section 15.5, "Chart Builder Enhancements"](#)

15.1 Changes to the Oracle BI Publisher Menu

In support of these new features, the Oracle BI Publisher menu contains the following new items:

Added to the Insert menu:

- **Repeating Group**
- **Conditional Format**
- **Conditional Region**

For information on these menu items, see [Section 15.4, "Advanced Form Field Support."](#)

Added to the Preview menu: **PowerPoint**

For information on this menu item, see [Section 15.3, "Microsoft PowerPoint Output Support."](#)

In addition, two menu items have been changed:

- **Update Layout Template is now Upload Template**
- **Publish Template As is now Upload Template As**

The functionality of these two menu items has not changed.

15.2 Integration with Oracle BI Discoverer

In this release, BI Publisher accepts Discoverer worksheets as report data sources. To facilitate building reports from the Discoverer worksheets, the Template Builder enables you to connect to Discoverer, browse worksheets, and load data to the Template Builder directly from Discoverer.

To open a Discoverer Report in the Template Builder:

Prerequisite: You must have set up integration with Discoverer in the BI Publisher Admin page. For more information see [Chapter 18, "Oracle BI Publisher Integration with Oracle BI Discoverer in Release 10.1.3.3"](#).

1. From the Oracle BI Publisher menu, select **Log On**.
2. Enter your credentials for Oracle BI Publisher. After successfully logging on, the Open Template dialog is displayed.
3. From the Workspace list, select **Discoverer**. The Select Data Connection dialog is displayed.
4. From the Select Data Connection dialog, select the connection to Discoverer.
5. Browse the Discoverer folders to locate the worksheet to open in the Template Builder. Double-click the worksheet. A dialog prompts you to Save As Oracle BI Publisher Report.
6. From the dialog, select the folder on the BI Publisher server where you want to save the BI Publisher report that uses this Discoverer worksheet as the data source.
7. Click **OK** to create a report in the folder. This also loads the report structure and data to the Template Builder.
8. You can now design the template using the full range of Template Builder features on the Discoverer data source.
9. When finished designing the layout template, save the template to the BI Publisher server by selecting **Upload Template** from the Oracle BI Publisher menu.

15.3 Microsoft PowerPoint Output Support

In this release, BI Publisher supports Microsoft PowerPoint as an output option. The Template Builder also supports PowerPoint as a preview option.

To preview a report in PowerPoint, select **PowerPoint** from the Oracle BI Publisher Preview menu.

This action launches Microsoft PowerPoint and displays the report as a PowerPoint presentation. Each report page is displayed on a new slide.

Note that to view BI Publisher PowerPoint output, you must have Microsoft PowerPoint 2003 or later installed on your computer.

For additional information regarding usage and limitations of the PowerPoint output feature, see [Chapter 17, "Oracle BI Publisher PowerPoint Output in Release 10.1.3.3."](#)

15.4 Advanced Form Field Support

In previous releases, BI Publisher inserted abbreviated XSL code into the Form Field Help Text Dialog. Long code strings required inserting code into both the Status Bar and Help Key tabs of the Form Field Help Text Dialog. These insertions made it challenging if you wanted to edit this code. With this release, you can edit code in the form fields in a single window in the BI Publisher Properties dialog. Further, several additional dialogs have been added to enable you to insert common format constructs in the template and the Template Builder generates the code for you.

The new dialogs simplify the insertion and editing of the following report elements:

- Data Field Properties
- Repeating Groups
- Conditional Formatting
- Conditional Regions

To open the Properties dialog for a field, perform one of the following:

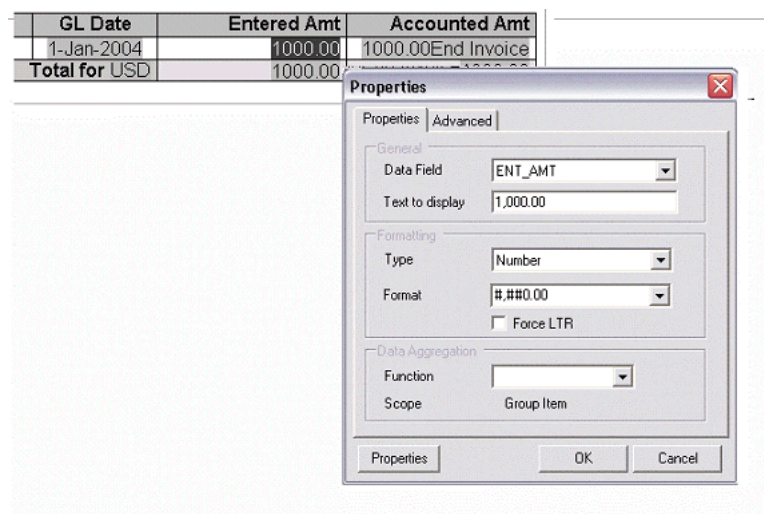
- Double-click the form field in the template.
- Right-click the form field, from the menu select BI Publisher and then select Properties.

For recognized code patterns (for example: data value, for-each, for-each-group, conditional formatting) the dialog displays a Properties tab and an Advanced tab (the Repeating Group dialog also displays a Sorting tab). The Properties tab displays modifiable properties for the field. The Advanced tab displays the underlying code. If the code pattern is not recognized, then the Properties dialog displays only the Advanced tab with the code from the form field.

Note that each dialog includes a Properties button that enables you to access the Microsoft Word Form Field Help Text dialog.

15.4.1 Edit Data Fields

The following screenshot displays the Properties dialog invoked for a data field:

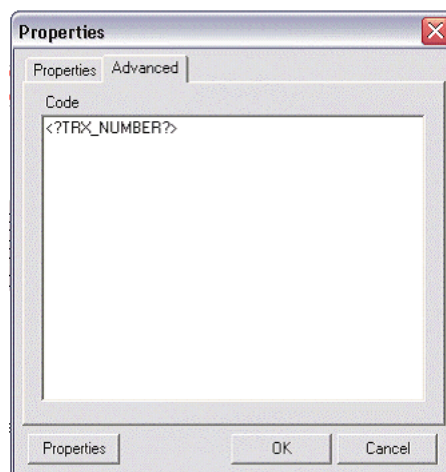


You can set the following properties for a data field:

- **Data Field** — Select the data field from the list of available fields from the loaded data source.
- **Text to Display** — Enter the display text for the form field in the template. This text is replaced at run time by the value in the data.
- **Type** — Select the type of data. Options are Regular Text, Number, Date, Current Date, Current Time. The selection in this field determines the format options.
- **Format** — For any data type except Regular Text, you can select from several number or date display formatting masks or enter your own.
- **Force LTR** — (Force Left-to-Right) This box is needed only if you are using the template in a language that prints the characters from right to left, such as Arabic or Hebrew. In these languages you might want to force left-to-right printing for fields such as phone numbers, addresses, postal/zip codes, or bank account number.
- **Function** — This feature enables you to perform aggregation functions (Sum, Average, Count, Minimum, Maximum) on data fields. For example, if you select sum for a data field, the field displays the sum of all occurring values for this data field depending on the scope, as described in the next item.
- **Scope** (informational only) — This field has two possible values:
 - **Group Item** — Indicates that the data field is inside a group. If you choose to perform a function on the field, then only the occurrences of the field within the current group are included in the aggregation.
 - **Normal** — Indicates that the field is not inside a group. Aggregation functions are performed on all occurrences of the field in the data.

15.4.2 Advanced Tab

The following screenshot shows the Advanced tab of the Properties dialog:



The Advanced tab displays the underlying code.

If the code pattern within the form field is not recognized, then Properties dialog includes this tab only. Use this tab to edit or add code to the form field. Select OK to update the template.

The Properties button opens the Microsoft Word Text Form Field Options dialog.

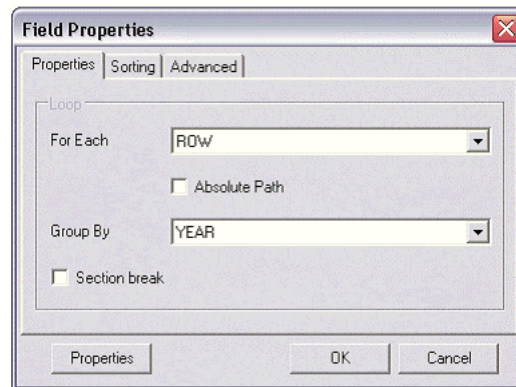
15.4.3 Insert and Edit Repeating Groups

The Template Builder now enables you to insert and edit for-each and for-each group fields using a Field Properties dialog. Use the dialog to choose the data elements to group and the data element to group them by.

15.4.3.1 Creating Repeating Groups

To create a repeating group:

1. Select the section of the template that contains the elements to repeat.
2. From the Oracle BI Publisher menu, select **Insert**, then **Repeating Group**.
3. Enter the appropriate fields in the Field Properties dialog:



- **For Each** — Select the repeating element to create the loop. When you select the For Each data field, you specify that for each occurrence of the field in the data, you want the elements and processing instructions contained within the loop to be repeated.

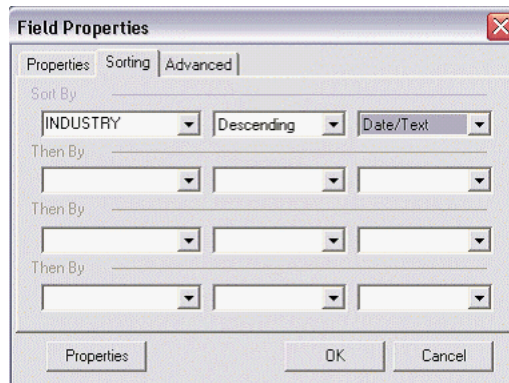
Note the following about creating repeating groups:

- For loops and groupings not inside another group (that is, outer groups or loops) you must select the repeating XML element to be used. For example if the data set is flat, then the only repeatable element is /DATA/ROWSET/ROW. In cases with multiple data sources or hierarchical XML you can choose the data set.
- If you are creating nested groups (inserting a loop or group inside of another loop in the template), then the For Each field is not updatable because it is defined by the preexisting outer loop. The For Each field displays "Group Item" to specify that an outer group is defined.
- **Absolute Path** — Select this box to use the Absolute Path to the element in the XML structure. This path is important if the data contains the same element name grouped under different parent elements.
- **Group By** — Select a field from the list by which you want to group the data. To create only a simple loop, do not select a group by element.
- **Section break** — Select this box to create a section break between each loop. Note the following characteristics of this feature:
 - The section break can be created only on outer loops that surround the entire document.
 - As a section break, the page numbering is reset, headers and footers are reset, and any running calculations are reset.

4. To sort the grouped data, select the Sorting tab. You can select up to four sort-by fields. For each sort by field, select the following:
 - **Sort order** — Select Ascending or Descending.
 - **Data type** — Select Number or Date/Text. It is important that you select the correct data type to achieve the expected sort order.

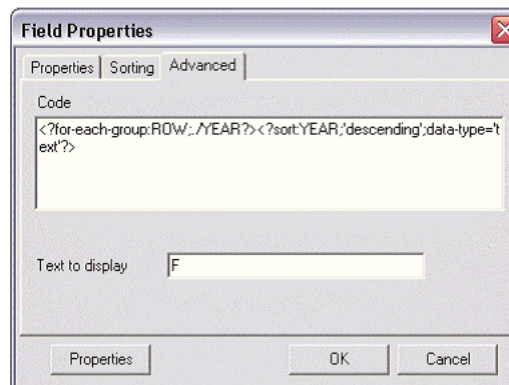
Note that if you are sorting by four criteria and the XML data element names are long, then you might exceed the character length limitation (393 characters) of the Microsoft Word form field.

The following figure shows the Sort tab of the Repeating Group Field Properties dialog:



5. To change the text display or edit the code directly, select the Advanced tab.

The following figure shows the Advanced tab of the Repeating Group Field Properties dialog:



6. When you have completed the dialog options, click **OK** to insert the form fields in the template. By default, the beginning for-each form field displays the text "F" and is inserted at the beginning of the selected template section. At the end of the selection, an "E" form field is inserted to denote the end of the repeating group.

15.4.3.2 Create Grouping Fields Around a Block

To create a group around an existing block of text in a template:

1. Select the block of text. For example, a table row.

Note that if any previously inserted BI Publisher tags are included in the block, then you must ensure that you include the beginning and ending tags. For example, if the block contains any opening for-each, if, or for-each-group tags,

then you must include the end for-each, end-if, and end for-each-group tags in the selection.

2. From the Oracle BI Publisher menu bar, select **Insert**, then **Repeating Group**.
3. In the Field Properties dialog, enter the fields to define the group as described in [Section 15.4.3.1, "Creating Repeating Groups."](#)
4. Click **OK** to insert the grouping fields around the block. For example, if the block is a table row, then the begin field is inserted at the beginning of the first cell and the end field is put at the end of the last field.

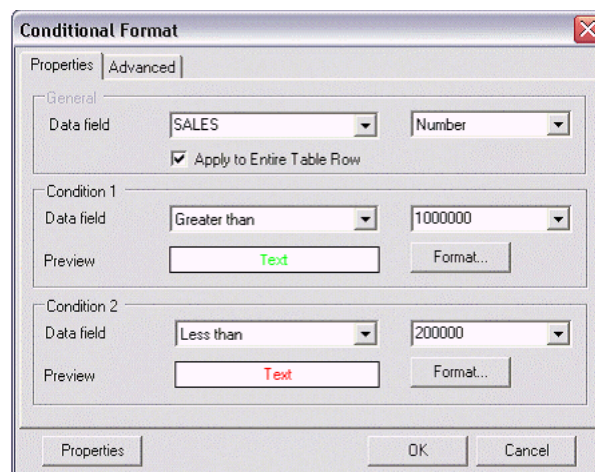
15.4.4 Insert Conditional Formatting

Using the Advanced Form Field features, you can insert simple conditional formats to apply to table rows or cells. The dialog provides several common options that you can select and the Template Builder inserts the code automatically. The Conditional Format dialog supports two conditions per field.

Use this feature to easily construct formats such as displaying values below a certain threshold in one format, and above the threshold in another.

To insert a conditional format:

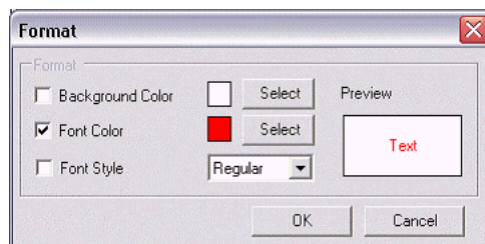
1. Place the cursor in the table cell of the data element for which you want to define the condition.
2. From the Oracle BI Publisher toolbar, select **Insert**, then **Conditional Format**.
3. Enter the following in the Conditional Format dialog:



- **Data Field** — Select the element to test for the condition and the data type of the element (Number or Date/Text).
- **Apply to Entire Table Row** — If you want the format applied to the entire table row, not just the cell of the selected element, then select this box.
- **(Condition 1) Data field** — Select the comparison operator. The options are:
 - Equal to
 - Not equal to
 - Greater than
 - Less than

- Greater than or equal to
- Less than or equal to
- Select the value to meet the condition. Note that you can enter an integer, enter text, or select another data element to define a comparison based on the incoming values.
- Click **Format** to define the format to apply when the condition is met. Options are background color, font color, and font style (regular, bold, italic, bold italic). Select the box and format of each option to apply. After you select the format, the Preview region displays the selected format.

The following figure shows the Format dialog from the Conditional Format dialog:



- Define a second condition if desired.
4. Select **OK**. The conditional format field is inserted as a form field with the display text "C".

To edit the conditional format, double-click the inserted form field to launch the dialog for editing; or, right-click the form field and select **BI Publisher**, then **Properties**.

15.4.5 Inserting and Editing Conditional Regions

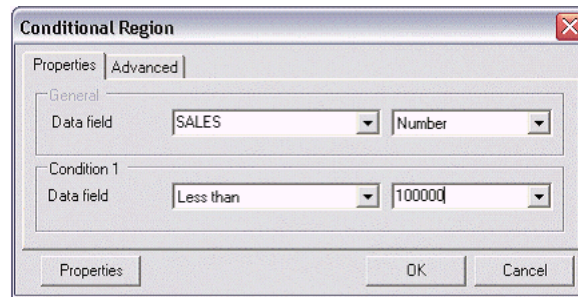
A conditional region is an area that is surrounded by a conditional statement. If the statement tests true, then the area is displayed in the report; if the condition tests false, then the area is suppressed from the report.

For example, suppose that the data contains salary and position information for all employees. The report contains a table that displays the employee, position, and salary. You want the report to display salary information for managers only. Using the insert conditional region functionality, you can select the region that contains the employee table and insert the condition that the position element must equal manager.

To insert a conditional region:

1. Select the region to apply the condition to. For example, if you want to display a table only for a certain condition, then select the region that contains the table. Note that the region must be inside a loop.
2. From the BI Publisher toolbar select **Insert**, then **Conditional Region**.

The following figure shows the Conditional Region dialog:



3. Enter the following fields:

- **Data field** - Select the field to test for the condition.
- Select the data type of the field: Number or Date/Text.
- Select the comparison operator for the data field. Options are:
 - Equal to
 - Not equal to
 - Greater than
 - Less than
 - Greater than or equal to
 - Less than or equal to
 - Enter the value to define the condition, or select from the list of data elements to define a conditional relationship based on the value of another data element.

4. Click **OK**. The form fields that contain the conditional logic are inserted around the region. The beginning form field displays the text "C" and the form field that closes the region displays the text "EC".

To edit the conditional region, double-click the inserted form field to launch the dialog for editing; or, right-click the form field and select **BI Publisher**, then **Properties**.

15.5 Chart Builder Enhancements

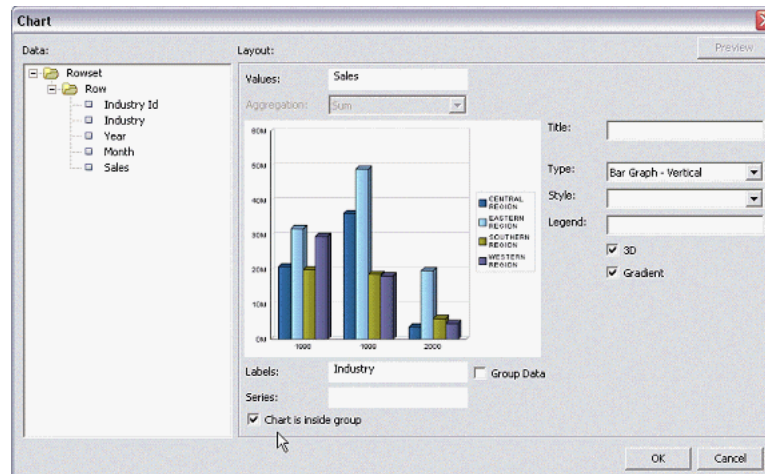
Enhancements to the Template Builder's chart builder include:

- Support for charts inside a grouping
- Support for new chart types
- Support for new chart subtypes

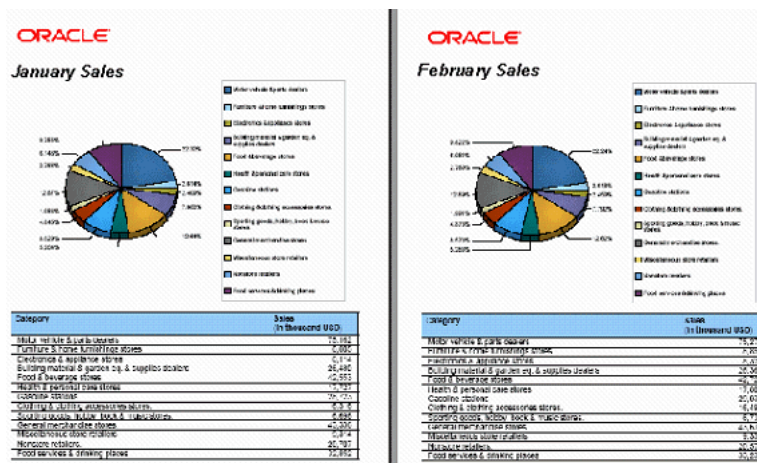
15.5.1 Support for Charts Inside a Grouping

You can now insert a chart inside a grouping of data and indicate through the chart builder to base the chart on the surrounding grouping.

The following figure shows the new box inside the chart builder dialog:



Using this feature, you can easily insert charts within groups to display the data for each grouping. For example, suppose that the report contains sales data. The template groups the data by year then by month to report the month's sales by industry. To insert a chart to display each month's data, simply insert the chart within the grouping tags and select the Chart is inside box. This selection enables you to create reports as shown in the following figure of a sample report with chart inside grouping:

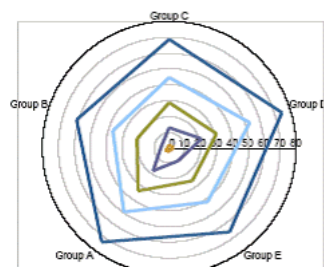


15.5.2 New Chart Types

The following chart types are now available as options from the chart builder:

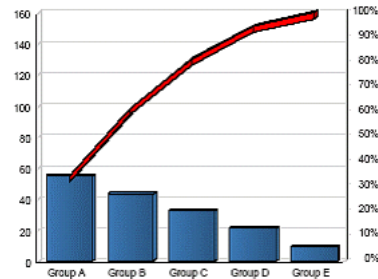
- **Radar**—Plots the same information as a bar graph, but displays the data radiating from the center of the graph.

The following figure shows a sample radar chart:



- **Pareto** — Displays criteria in descending order to summarize the relative importance of the differences between groups of data. In this graph type, a line shows a cumulative total of the percentages. When using this type of chart, ensure that you sort the data numerically, descending by value (not label).

The following figure shows a sample pareto chart:



15.5.3 New Chart Subtypes

The chart builder now supports additional chart subtypes in this release. Stacked and percentage subtypes are now supported for bar charts, line charts, and area charts. These are displayed as follows on the chart type list:

- Bar - Vertical - Percent
- Bar - Vertical - Stacked
- Bar - Horizontal - Percent
- Bar - Horizontal - Stacked
- Line - Percent
- Line - Stacked
- Area - Percent
- Area - Stacked

Flash Templates for Oracle BI Publisher in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

This new feature enables report authors to develop Adobe FLEX templates that can be applied to BI Publisher reports to generate highly interactive Flash output documents.

Beginning in Release 10.1.3.3, BI Publisher offers support for Adobe Corporation's new document format for building interactive forms and reports, called Flex. You can build Flex templates, test them on your desktop, and deploy them to the BI Publisher server to generate Flash output. Users can then run the reports from the BI Publisher user interface or schedule them for delivery to report consumers.

This chapter describes how to set up a Flex template with a BI Publisher "flat" data source (that is, there is no hierarchy in the XML data) and describes the simpler objects such as tables and charts. For more information about interactivity, connectivity between components, and more advanced topics, refer to Adobe's Flex documentation.

This chapter includes the following sections:

- [Section 16.1, "Changes to the BI Publisher User Interface"](#)
- [Section 16.2, "Prerequisites"](#)
- [Section 16.3, "Building Templates"](#)
- [Section 16.4, "Loading Templates"](#)
- [Section 16.5, "Setting Properties for PDF Output"](#)
- [Section 16.6, "Summary"](#)

16.1 Changes to the BI Publisher User Interface

To support Flash templates, the following changes are made in the BI Publisher user interface:

- On the Edit Report page, under Layouts, General Settings, the Template Type list includes the option **Flash Template**.
- On the View Report page, the output selection list includes the option **Flash** when a Flash Template is selected as the Template Type. Flash templates support only these output types: Flash, PDF, and data.

- A new set of "Flash" properties has been added to the Runtime Configuration page.

16.2 Prerequisites

The prerequisites are as follows:

- For viewing output:
 - To view the report output from the Flash Template, you must have Adobe Flash Player 9 installed on your computer.
- For building templates:
 - The FlexBuilder IDE from Adobe. BI Publisher is currently certified with version 2.0.1. The tool can be downloaded and purchased from the Adobe Web site - <http://www.adobe.com/products/flex/>.
Note that the charting functionality is available for an additional license fee.
 - A report data model set up in BI Publisher that generates flat XML.

16.3 Building Templates

This section includes the following topics:

- ["Adding the Data Source"](#)
- ["Creating the Layout"](#)
- ["Data Binding"](#)

16.3.1 Adding the Data Source

1. Generate a sample data file from the report data model. From the BI Publisher Report Editor or from the Reports page, select View. If no layouts are defined for the report, then the output type defaults to xml. Otherwise, select Data for the output type. Select Export and save the results as an XML file to a local directory. This example is based on the following data:

```
<ROWSET>
<ROW>
<NAME>Neena Kochhar</NAME>
<FIRST_NAME>Neena</FIRST_NAME>
<LAST_NAME>Kochhar</LAST_NAME>
<SALARY>17000</SALARY>
<ANNUAL_SALARY>204000</ANNUAL_SALARY>
<FED_WITHHELD>57120</FED_WITHHELD>
<JOB_TITLE>Administration Vice President</JOB_TITLE>
<DEPARTMENT_NAME>Executive</DEPARTMENT_NAME>
<MANAGER>Steven King</MANAGER>
</ROW>
<ROW>
...
</ROWSET>
```

This data is generated from the following simple query-based report:

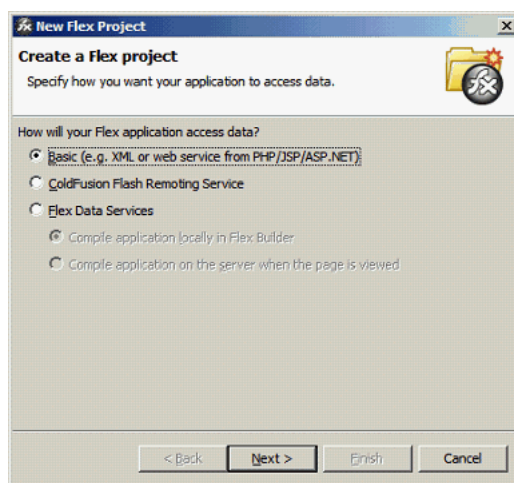
```
select
    e.first_name || ' ' || e.last_name name,
    e.first_name,
    e.last_name,
```

```

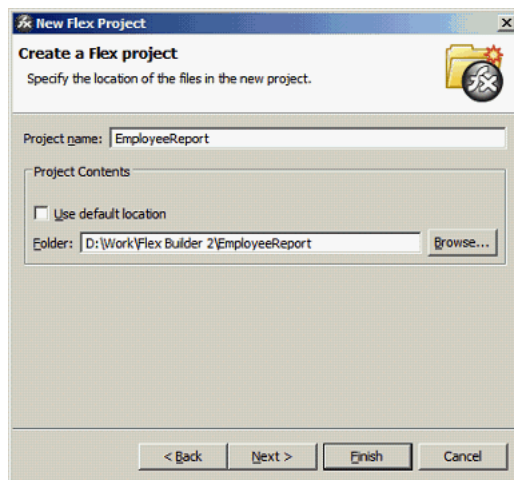
        e.salary,
        e.salary*12 ANNUAL_SALARY,
        e.salary*12*0.28 FED_WITHHELD,
        j.job_title,
        d.department_name,
        m.first_name || ' ' || m.last_name manager
from employees e,
     employees m,
     departments d,
     jobs j
where e.department_id = d.department_id
     and j.job_id = e.job_id
     and e.manager_id = m.employee_id

```

2. Open the Flex IDE and create a new Flex Project, select the "Basic" data access method.



On the next screen specify a name for the project. The name that you use here is assigned to the template file name that you are going to create.



Click **Finish**.

The IDE creates an MXML file, which is the Flex template definition file. The file has an XML format as follows:

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
</mx:Application>
```

You can now update it manually or by using the visual builder.

3. Connect the XML that you downloaded from the report data model. Use the XML data services that Flex supports to embed the sample data into the MXML file.

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute">
  <mx:Script>
    <![CDATA[
      [Bindable]
      public var dataXML:XML =
<ROWSET>
<ROW>
<NAME>Neena Kochhar</NAME>
<FIRST_NAME>Neena</FIRST_NAME>
<LAST_NAME>Kochhar</LAST_NAME>
<SALARY>17000</SALARY>
<ANNUAL_SALARY>204000</ANNUAL_SALARY>
<FED_WITHHELD>57120</FED_WITHHELD>
<JOB_TITLE>Administration Vice President</JOB_TITLE>
<DEPARTMENT_NAME>Executive</DEPARTMENT_NAME>
<MANAGER>Steven King</MANAGER>
</ROW>
<ROW>
...
</ROWSET>;
    ]]>
  </mx:Script>
</mx:Application>
```

The XML portion should look familiar as the data that you downloaded. The important components to note are:

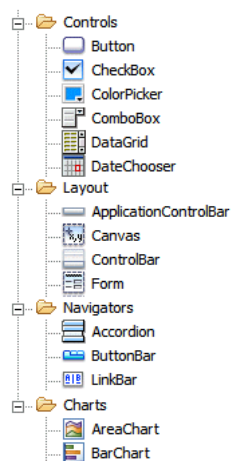
- `<mx:Script>` — This denotes the start of the template scripting code. There is also a closing `</mx:Script>` statement.
- `[Bindable]` — This denotes that the following variable is bindable to a layout component.
- `public var dataXML:XML` — This is the data variable declaration:
 - `public` — The value of the variable is available to the whole template.
 - `var` — Declares there is a variable in the report.
 - `dataXML` — The name of the variable. Note this is a compulsory name. You must use this name to use the template with BI Publisher.
 - `:XML` — Declares that the variable is an XML type.
- `;` — Notice the semicolon after the end of XML data you provided.

At run time the BI Publisher server generates the run-time data from the report and injects the data into the Flex template, replacing the sample data held within the `dataXML` variable. This feature allows the Flex report to be distributed to users without needing to connect to the server.

16.3.2 Creating the Layout

The next step is to add report objects to the layout palette. This example uses the Flex Design tab to add the objects to the layout. Click the Design tab to see the available objects in the Component Navigator pane.

The following figure shows an example of the available objects in the Component Navigator pane:



These objects can be dragged and dropped to the design palette.

1. The Flex IDE creates a default canvas for you to drop objects onto. You can modify the canvas as required.

IMPORTANT: If you intend to embed the Flash output in a PDF document, then you must set the Width and Height of the template in the Size region of the Layout properties. To accept the default size, you must explicitly enter values in these fields.
2. Start by dragging a Panel object from under the Layout node to the design palette. Notice as you drag the panel around the edge of the palette that you see guidelines displayed in blue. Use these guides to aid you in aligning objects.
3. Drop the panel onto the top left hand corner of the palette.
4. Now drag the bottom right edge of the panel across to the right-hand side of the palette.
5. Then drag it down to about half the height of the palette. Alternatively, use the property palette on the right hand side to set the size of the panel.
6. Pick up a Datagrid object. This is the object to render the data in a tabular format. Drop it onto the panel that you created in Step 2. The Datagrid is now a child of the panel; you can resize it as needed. The result is shown in the following figure:

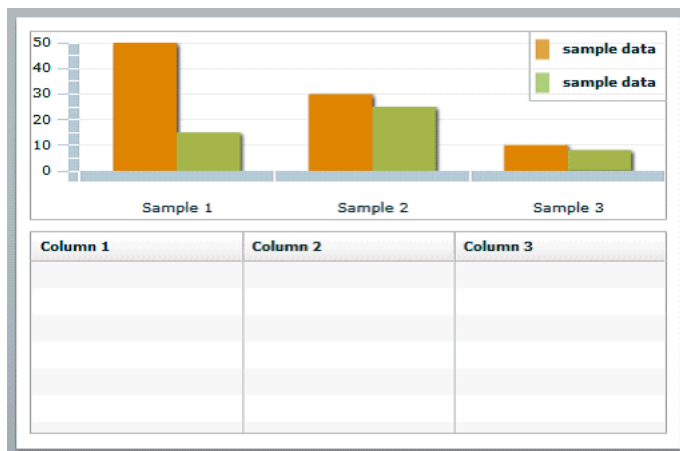
Column 1	Column 2	Column 3

By default three columns are generated. In the next section, you override the default in the MXML code.

16.3.2.1 Adding a Chart

If you have purchased the charting option, then you can add charts to the layout.

1. Make room for the chart in the layout. Highlight the Datagrid object and pull the top edge down to about half the height of the hosting panel.
2. For the purposes of this guide, select and drag a Column Chart from the design palette and drop it onto the hosting panel. Use the guidelines to get it aligned.
3. When you drop it, you see that the default size overlaps the Datagrid and that the chart legend is in the top left-hand corner. Resize the chart and move the legend to the left to look similar to the following figure:



This is a sample chart. You can bind it to the data in the next section.

16.3.3 Data Binding

Now that the layout is complete, you can bind the layout objects to the data source. Flex does offer some help through the property palette of the objects to define the binding, but not enough to complete the task. Therefore the following steps describe how to update the MXML directly using the "Source" editor.

16.3.3.1 Binding the DataGrid

To bind the DataGrid:

1. Start by highlighting the Datagrid in the design palette, and click the Source tab to display the MXML source. You see that the first line of the Datagrid code has been highlighted. This highlighting is a useful feature if you have built complex Flex templates and must locate the code easily.

The Datagrid code is as follows:

```
<mx:DataGrid x="10" y="160" width="476" height="152">
  <mx:columns>
    <mx:DataGridColumn headerText="Column 1" dataField="col1"/>
    <mx:DataGridColumn headerText="Column 2" dataField="col2"/>
    <mx:DataGridColumn headerText="Column 3" dataField="col3"/>
  </mx:columns>
</mx:DataGrid>
```

Notice that it defines the relative x,y position of the grid within its parent container and its width and height. The next element defines the columns with attributes for the header label and the data fields.

The goal is to achieve a table that looks like the following figure:

Employee	Title	Monthly Salary	Annual Salary
Neena Kochhar	Administration Vice	17000	204000
Lex De Haan	Administration Vice	17000	204000
Alexander Hunold	Programmer	9000	108000
Bruce Ernst	Programmer	6000	72000

2. Make the Datagrid aware of the data source. To do that, add an attribute to the `<mx:DataGrid>` element as follows:

```
dataProvider="{dataXML.ROW}"
```

This attribute defines the data object to be used at run time to populate the grid. Remember that the XML data variable was initially defined as "dataXML"; use that followed by "ROW". ROW is the repeating group in the data set. Include the curly braces to specify for the Flex engine about the data source definition.

3. Bind the columns by replacing values as shown in the following example:

```
<mx:columns>
  <mx:DataGridColumn headerText="Employee" dataField="NAME" />
  <mx:DataGridColumn headerText="Title" dataField="JOB_TITLE"/>
  <mx:DataGridColumn headerText="Monthly Salary" dataField="SALARY"/>
  <mx:DataGridColumn headerText="Annual Salary" dataField="ANNUAL_SALARY"/>
</mx:columns>
```

4. With this update, review the following definition for the datagrid:

```
<mx:DataGrid x="10" y="160" width="476" height="152"
dataProvider="{dataXML.ROW}">
  <mx:columns>
    <mx:DataGridColumn headerText="Employee" dataField="NAME" />
    <mx:DataGridColumn headerText="Title" dataField="JOB_TITLE"/>
    <mx:DataGridColumn headerText="Monthly Salary" dataField="SALARY"/>
    <mx:DataGridColumn headerText="Annual Salary" dataField="ANNUAL_SALARY"/>
  </mx:columns>
```

```
</mx:DataGrid>
```

5. Test the template with the sample data. Use the Run > Run EmployeeReport. This command opens a new browser window and renders the table with the sample data as described in Step 1.

16.3.3.2 Binding the Chart

To bind the chart:

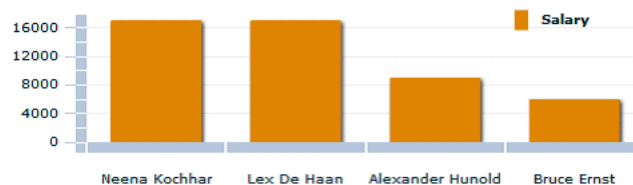
1. Start by displaying the Design tab and highlighting the Chart, then flip back to the Source view to find the chart code:

```
<mx:ColumnChart x="10" y="10" id="columnchart1" width="476" height="142">
  <mx:series>
    <mx:ColumnSeries displayName="Series 1" yField="/">
    </mx:series>
</mx:ColumnChart>
<mx:Legend dataProvider="{columnchart1}" x="383" y="10"/>
```

2. To bind the data source to the chart object, add the dataProvider attribute to the <mx:ColumnChart> as follows:

```
dataProvider="{dataXML.ROW}"
```

3. Add the binding for the horizontal axis and the column series. Check the Flex help files for more details. Suppose that you need to create a chart that shows salaries by employees, similar to the following one:



To achieve this format, modify the <series> group and add a <horizontalAxis> element, as shown in this example:

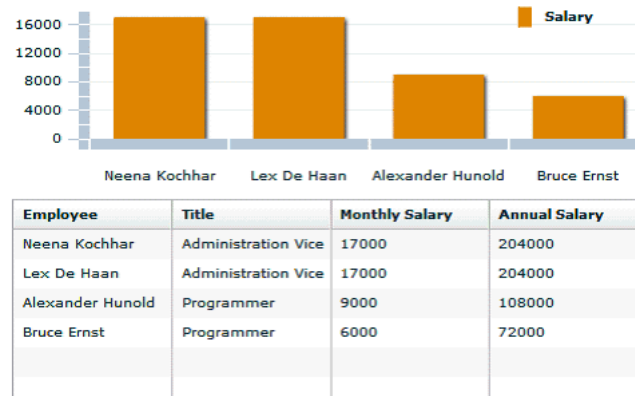
```
<mx:ColumnChart x="10" y="10" id="columnchart1" width="476" height="142"
dataProvider="{dataXML.ROW}">
  <mx:horizontalAxis>
    <mx:CategoryAxis categoryField="NAME" />
  </mx:horizontalAxis>
  <mx:series>
    <mx:ColumnSeries xField="NAME" yField="SALARY" displayName="Salary"/>
  </mx:series>
</mx:ColumnChart>
<mx:Legend dataProvider="{columnchart1}" x="383" y="10"/>
```

The <mx:horizontalAxis> element has been added and the categoryField attribute has the NAME data element assigned. This element is required to render the chart.

The <mx:series> element has been updated binding the SALARY value to each of the employee NAMES to create a bar for each employee.

The legend code does not need updating. Notice the "id" attribute on the ColumnChart matches the dataProvider attribute value on the <mx:Legend> element.

4. You can now run the template using the sample data. The chart is displayed above the tabulated data, as shown in the following figure:



16.4 Loading Templates

After testing the Flex template on the desktop, you can load it to the BI Publisher server.

1. Using the Admin or Developer role, navigate to the Edit Report page.
2. Highlight the Layouts node and use the Upload Template field to upload the Flex template to the server. To locate the correct file, use the Browse button and navigate to the Flex project directory. Under this directory open the bin directory and select the EmployeeReport.swf file. Use the Upload button to load the template.
3. With the Layouts node still highlighted, click New at the top of the report structure tree. Enter the name, template and template type fields as shown in the following figure.

4. Save the report. The template is now available for users to select to either run in real time or to schedule.

16.5 Setting Properties for PDF Output

A new set of Flash properties has been added to the Runtime Configuration page to enable you to specify the size and placement of the Flash object when you select PDF as the output type.

IMPORTANT: To produce PDF output, you must specify the height and width of the template in the Flex Builder. See [Section 16.3.2, "Creating the Layout."](#)

To configure the PDF output:

1. Using the Administrator or Developer role, navigate to the report to use the Flash template and click Configure.

On the Runtime Configuration page, you see a new set of properties under the Flash heading. These properties control the placement and sizing of the Flash object in the output PDF document. The properties are shown in the following figure:

Flash		
Page width of wrapper document	<input type="text"/>	792 (in points, 11 inch)
Page height of wrapper document	<input type="text"/>	612 (in points, 8.5 inch)
Start x position of Flash area in PDF	<input type="text"/>	18 (in points, 0.25 inch)
Start y position of Flash area in PDF	<input type="text"/>	18 (in points, 0.25 inch)
Width of Flash area	<input type="text"/>	Same as flash width in points in swf
Height of Flash area	<input type="text"/>	Same as flash height in points in swf

2. Enter values for the properties. Note that no properties are required. If you do not enter any values, then the default values assume an 11 inch by 8.5 inch document (standard landscape), with a quarter inch inset from the upper-left corner of the page as the insertion point of the Flash object. The default area in the document is the size of the SWF object.
 - Page width of wrapper document – Specify in points the width of the output PDF document. The default is 792, or 11 inches.
 - Page height of wrapper document – Specify in points the height of the output PDF document. The default is 612, or 8.5 inches.
 - Start x position of Flash area in PDF – Using the left edge of the document as the 0 axis point, specify in points the beginning horizontal position of the Flash object in the PDF document. The default is 18, or .25 inch.
 - Start y position of Flash area in PDF – Using the upper-left corner of the document as the 0 axis point, specify in points the beginning vertical position of the Flash object in the PDF document. The default is 18, or .25 inch.
 - Width of Flash area – Enter in points the width of the area in the document for the Flash object to occupy. The default is the width of the SWF object.
 - Height of Flash area – Enter in points the height of the area in the document for the Flash object to occupy. The default is the height of the SWF object.

16.6 Summary

This section has covered the basics of adding and organizing layout objects, binding the objects to BI Publisher data sources to create Flex templates, and loading the templates to the BI Publisher server to make them available to report users.

This section has been a basic introduction to Flex templates but Adobe Flex enables you to build far more complex interactive reports for users. The animation, "wiring," and formatting of layout objects can be achieved with Flex. You can also summarize and create calculated fields on the incoming data. See the Flex documentation for these more advanced features.

Oracle BI Publisher PowerPoint Output in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

Release 10.1.3.3 adds support for Microsoft PowerPoint output. This support enables you to get report data into your key business presentations.

In this initial version of this functionality, the PowerPoint document that is generated is a simple export of the formatted data and charts to PowerPoint. A future release provides more sophisticated integration that allows for direct embedding and refreshing from within PowerPoint.

This chapter includes the following sections:

- [Section 17.1, "Supported Versions of PowerPoint"](#)
- [Section 17.2, "Configuring Fonts for BI Publisher Server"](#)
- [Section 17.3, "Configuring Fonts for BI Publisher Template Builder"](#)
- [Section 17.4, "Usage and Limitations"](#)

17.1 Supported Versions of PowerPoint

The supported versions of PowerPoint are:

- Microsoft Office PowerPoint 2003
- Microsoft Office PowerPoint 2007

17.2 Configuring Fonts for BI Publisher Server

Support for PowerPoint output does not include the font fallback mechanism that is used for other types of output in BI Publisher. On the BI Publisher server, you must configure each font that is used in the RTF template for generating PowerPoint output. You must copy these fonts to the BI Publisher Server and define the Font Mappings for RTF templates. These tasks can be done for the entire system or for individual reports.

See *Oracle Business Intelligence Publisher Report Designer's Guide* section "Defining Font Mappings" for more details:

http://download.oracle.com/docs/cd/B40078_02/doc/bi.1013/b40017/T421739T421745.htm#453835

17.3 Configuring Fonts for BI Publisher Template Builder

To correctly preview PowerPoint output that uses non-English or nonstandard fonts, you must define the fonts in the BI Publisher configuration file. This configuration file is called `xdo.cfg` and is typically found in the following location:

`C:\Oracle\BI Publisher\BI Publisher Desktop\Template Builder for Word\config\`

Note that if you have not used this file yet, you might find the file "`xdo example.cfg`" instead. This file must be saved with an encoding of UTF-8 and provide a full and absolute path for each font defined. Otherwise, you might encounter issues such as characters overlays and wrapping that does not work.

To summarize, if you are using a font that is not defined in `xdo.cfg` it might not display correctly in the generated Microsoft Office PowerPoint file. To correct this situation:

1. Add these font items into the `xdo.cfg` file, and specify the absolute path of the font files locations.
2. Save the `xdo.cfg` file in UTF-8 format.
3. Ensure that all the fonts used in the template file are defined correctly in the `xdo.cfg` file.

The following figure shows an example of a `xdo.cfg` file:

```
<config version="1.0.0" xmlns="http://xmlns.oracle.com/oxp/config/">
<!-- Properties -->
<properties>
</properties>

<!-- Font setting -->
<font>
  <font family="宋体" style="" weight="">
    <truetype
      path="C:\Program Files\Oracle\BI Publisher\BI Publisher Desktop\Template Builder for Word\fonts\simsun.ttc"/>
    </truetype>
  </font>
  <font family="Times New Roman" style="" weight="">
    <truetype
      path="C:\Program Files\Oracle\BI Publisher\BI Publisher Desktop\Template Builder for Word\fonts\times.ttf"/>
    </truetype>
  </font>
</font>
</config>
```

17.4 Usage and Limitations

PowerPoint output is supported for RTF templates. See the *Oracle Business Intelligence Publisher Report Designer's Guide* section "Creating an RTF Template" for details on how to create an RTF template.

Adhere to the following guidelines to effectively use the templates for PowerPoint:

- Page breaks are treated as new slides.
- Background color of generated PowerPoint file is always white.
- Table border type must be single line. (Double border, dash, and other types are not supported.)
- Hyperlinks are not supported.
- Charts and Graphs are generated as PNG images. (SVG is not supported.)
- Shapes are not supported.
- Text position might be slightly incorrect if you use right-align.

- PowerPoint output preserves the page orientation (portrait or landscape) that is defined in the RTF template. Most presentations are oriented in landscape so this is the recommended orientation of an RTF template.
- Paper size must be the same on all pages of the RTF template. You cannot have mixed paper sizes in the same document.
- All Unicode languages, except bi-directional languages, are supported.
- Text position might be slightly incorrect for Chinese, Japanese, and Korean fonts when using bold or italic effects on characters. This is because Microsoft has bold/italic emulation when there is no bold/italic font.

Once you have uploaded an RTF template to a report, you have the option to determine if PowerPoint is a suitable output format for the template by selecting the **Limit Output Format to** button and selecting or deselecting the box next to **Powerpoint**. See the following figure.

The screenshot shows the 'Layout' tab with the 'General Settings' section. The 'Name' field is 'default'. The 'Template' dropdown is 'default.rtf' with a 'Refresh' button. The 'Template Type' dropdown is 'RTF Template'. Under 'Output Format', the 'Limit Output Format to' radio button is selected. Below it, several checkboxes are listed: HTML (checked), PDF (checked), RTF (unchecked), Excel (unchecked), Powerpoint (checked), CSV (checked), and Data (unchecked).

You can then set the output to **PowerPoint** when viewing the report.

The screenshot shows the report viewer interface. The 'Template' dropdown menu is open, showing options: HTML, PDF, PowerPoint (highlighted), and CSV. To the right of the dropdown are buttons for 'View', 'Export', 'Send', and 'Schedule'. Below the dropdown, the word 'DATA' is visible.

Oracle BI Publisher Integration with Oracle BI Discoverer in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

BI Publisher uses a modern report architecture that separates the report data model from the report layout. By doing so, it allows users to create multiple layouts for a single data model. BI Publisher also supports many types of data sources such as SQL query against a database using JDBC, file data, HTTP feed, and Web service.

Using the Discoverer Web service, you can create reports based on the data that is returned from Discoverer worksheets. When you install the Discoverer Web service and configure BI Publisher to use it, the Web service returns data from selected worksheets that provide the data for BI Publisher reports.

BI Publisher 10.1.3.3 includes the ability to use Discoverer worksheets as the data set for a report. This provides the following three key benefits to Discoverer users:

- Create highly formatted reports
- Schedule execution and delivery of reports to a wide range of destinations
- Generate reports in a greater range of formats

This chapter includes the following sections:

- [Section 18.1, "Configuring the Oracle BI Publisher and Discoverer Integration"](#)
- [Section 18.2, "Creating Reports Based on Discoverer Worksheets"](#)

18.1 Configuring the Oracle BI Publisher and Discoverer Integration

The process to configure the Oracle BI Publisher and Discoverer integration includes the following steps:

[Step 1: Install the Discoverer Web Service](#)

[Step 2: Install Oracle BI Publisher 10.1.3](#)

[Step 3: Configure Discoverer to Use Oracle Application Server Identity Management](#)

[Step 4: Set Up a Dedicated User to Access the Discoverer Web Service from Oracle BI Publisher](#)

[Step 5: Configure Oracle BI Publisher to Use the Discoverer Web Service](#)

18.1.1 Step 1: Install the Discoverer Web Service

Install Application Server Patch Set 10.1.2.3 patch or the Discoverer one-off patch that includes the Discoverer Web service. This installs and deploys the Discoverer Web service file (discWS.jar) into the same Oracle Containers for J2EE (OC4J) that hosts Discoverer services. After installation, the Discoverer Web service is accessible from:

`http://<1012AppServer>:<port>/discoverer/wsi`

18.1.2 Step 2: Install Oracle BI Publisher 10.1.3

Release 10.1.3.3 of BI Publisher supports access to Discoverer worksheets as data sets in BI Publisher reports. BI Publisher and Discoverer can be, and usually are on separate computers.

18.1.3 Step 3: Configure Discoverer to Use Oracle Application Server Identity Management

To configure Discoverer to use Oracle Application Server Identity Management, see *Oracle Business Intelligence Discoverer Configuration Guide*, Section 14.7 "Using Discoverer with Oracle Identity Management Infrastructure," which is available from the Documentation pages on the Oracle Technology Network (OTN) or directly at:

http://download.oracle.com/docs/html/B13918_03/security2.htm#i1021521

18.1.4 Step 4: Set Up a Dedicated User to Access the Discoverer Web Service from Oracle BI Publisher

The Discoverer Web service is accessible only to trusted clients. The Administrator must set up the dedicated user that Oracle BI Publisher uses to access the Web service.

To set up the user:

1. Log in to an Application Server Control instance that has access to the Discoverer middle tier computer.
2. Select the Discoverer middle tier computer.
3. Select **OC4J_BI_FORMS**.
4. Select the **Applications**.
5. Select the deployed application named 'discoverer'.
6. Under the Administration section, choose Security.
7. Click the **Add User** button.

ORACLE Enterprise Manager 10g
Application Server Control

Log Log Topology Preferences Help

Farm > Application Server: as101202_MT.xmlps2.us.oracle.com > OC4J: OC4J_BI_Forms > Application: discoverer > Security >

Security: Add User

General

Name	bipublisher
Description	For BI Publisher to connect to BI Discoverer Web Service
Password	*****
Confirm Password	*****

8. Map this user to security role sr_users by selecting the sr_users Security Role and clicking on the "Map Role to Principals" button. On the Role: sr_users page, check

the box next to the newly created user in the "Map Role To Users" section and click **Apply**.

The screenshot shows the Oracle Enterprise Manager 10g Application Server Control interface. The breadcrumb trail is: Farm > Application Server: as101202_MT.amps2.us.oracle.com > OC4J: OC4J_BI_Forms > Application: discoverer > Security >. The page title is "Role: sr_users" and it indicates "Page Refreshed Jul 2, 2007 9:17:43 PM".

Under the "Map Role To Groups" section, there are links for "Select All" and "Select None". A table lists groups with checkboxes: "DiscoVWS/trustedusers" is unchecked, and "DiscoVWS/wsusers" is checked.

Under the "Map Role To Users" section, there are also links for "Select All" and "Select None". A table lists users with checkboxes: "DiscoVWS/bipublisher" is checked, "DiscoVWS/nls_user" is unchecked, "DiscoVWS/sba" is unchecked, "DiscoVWS/trustedsba" is checked, "DiscoVWS/trustedxmipublisher" is unchecked, "DiscoVWS/xmipublisher" is unchecked, and "DiscoVWS/用户" is unchecked.

At the bottom right, there are "Revert" and "Apply" buttons.

The role `sr_users` is pre-defined in the Discoverer `web.xml` file.

For more information, refer to "Security Role Mapping" in *Oracle Application Server Containers for J2EE Security Guide 10g Release 2 (10.1.2)*, which is available from the Documentation pages on the Oracle Technology Network (OTN) or directly at:

http://download.oracle.com/docs/cd/B14099_19/web.1012/b14013/jaas_integrate.htm#sthref270

18.1.5 Step 5: Configure Oracle BI Publisher to Use the Discoverer Web Service

Configure the Discoverer Web service in Oracle BI Publisher Admin pages using the Discoverer Web service URL from [Step 1: Install the Discoverer Web Service](#) (`http://<1012AppServer>:<port>/discoverer/wsi`) and the dedicated user that you defined in [Step 4: Set Up a Dedicated User to Access the Discoverer Web Service from Oracle BI Publisher](#).

The screenshot shows the Oracle BI Publisher Admin console with the "Admin" tab selected. The breadcrumb trail is: Admin > Oracle BI Discoverer. The "Integration" section is active, showing "Oracle BI Presentation Services" and "Oracle BI Discoverer".

A tip states: "Any changes will only take effect after the application is restarted." There are "Cancel" and "Apply" buttons.

The configuration fields are:

- Server Protocol: `http` (dropdown)
- * Server: `myserver.company.com` (text field)
- * Port: `7779` (text field)
- * Administrator Username: `bipublisher` (text field)
- * Administrator Password: `*****` (password field)
- URL Suffix: `discoverer/wsi` (text field, with default value `discoverer/wsi`)
- Fetch Size: `250` (text field, with default value `5000`)

Note that in this release, any Oracle BI Publisher server or cluster of servers can be configured with only one Discoverer instance.

Restart the application server that is running Oracle BI Publisher to allow these changes to take effect.

18.1.6 Discoverer Web Service Configuration Settings

The following configuration parameters that control the behavior of the Discoverer Web service can be set in the Discoverer configuration.xml file:

- **Loglevel** — Can be set to any one of following: WARNING, DEBUG, FATAL, ERROR, TRACE, NOTIFICATION.
- **MaxSessionPoolSize** — Determines the maximum number of Discoverer sessions that can be active and available at the same time.

These should be changed only under the direction of Oracle Support.

18.2 Creating Reports Based on Discoverer Worksheets

BI Publisher's modern report architecture provides benefits in report development and maintenance.

With this architecture, it is possible to have IT staff focus on extracting exactly the needed data with minimal impact on transactional systems. Then end users or IT staff can easily create as many layouts as needed for each query or data model. Layout development can even be done offline to further insulate transactional systems. Likewise it is a very straightforward process to create translations as needed for each layout.

18.2.1 About Creating a Report Based on a Discoverer Worksheet

In Oracle BI Publisher, each report has a data model that can be comprised of one or more data sets. Each data set is based on a data source that is defined in the Admin page.

For integration with Discoverer, the data source is configured through the Oracle BI Discoverer tab on the Integration page from the Oracle BI Publisher Admin page (see [Section 18.1.5, "Step 5: Configure Oracle BI Publisher to Use the Discoverer Web Service"](#)). When you want to create a report that uses data from Discoverer, you identify a Discoverer connection and select a worksheet that defines the data set. **Note:** A data model can include only one data set based on a Discoverer Worksheet.

There are two ways to create an Oracle BI Publisher report that uses a Discoverer Worksheet as a data set. One way is to create the report directly through the Oracle BI Publisher web interface. The other way is to indirectly create a report and data set while creating a layout template using the Oracle BI Publisher Template Builder for Word.

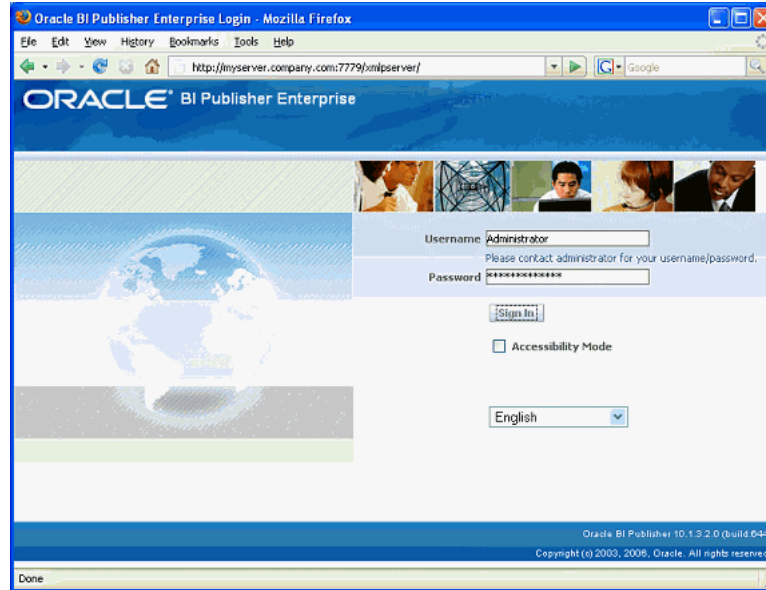
- Creating a report directly through the Oracle BI Publisher Web interface gives you control over the naming of the report and offers the option of creating sample data for layout development. This is helpful when the Discoverer Worksheet returns substantial amounts of data or has parameters with default values that return no data. In general, creating a report directly through the Web interface is the preferred method of creating the report and data model.
- You can also create a report indirectly through the Oracle BI Publisher Template Builder add-in to Word. This method creates a report with a name that is of the form workbook.worksheet in the directory of your choosing in the Oracle BI Publisher report repository. This method does not allow you to change or set default parameter values or to limit the amount of data that is returned from the Discoverer worksheet. This method is recommended only if the Discoverer worksheet returns a relatively small amount of data and does not have default parameter values that return no data. If the worksheet returns too much data, then

it overwhelms the Template Builder. See [Section 18.2.4, "Prerequisites, Limitations, and Known Issues."](#) Conversely, if the worksheet has parameters with default values such that no data is returned, then no sample data is available to build the layout template.

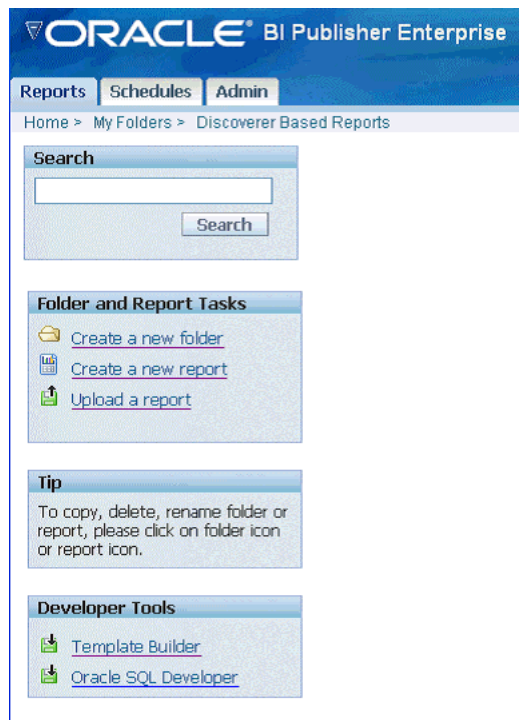
18.2.2 Creating a Report Directly Through the Oracle BI Publisher Web Interface

To create a report and data model based on a Discoverer worksheet through the Web interface to the Oracle BI Publisher server:

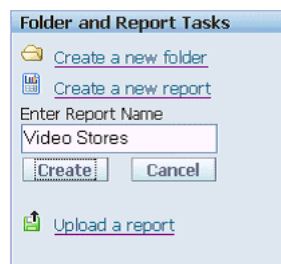
1. Connect to Oracle BI Publisher through the browser (for example, `myserver.company.com:7779/xmlpserver/`)



2. Navigate to or make a directory to hold the report to create.



3. Click **Create a new report** in the Folder and Report Tasks frame on the left side of the browser page. Provide a name and click **Create**.

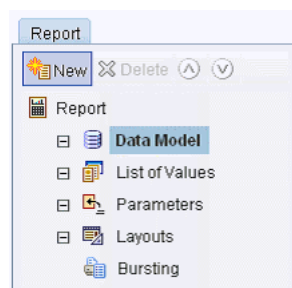


The report is displayed on the right-hand side of the screen.

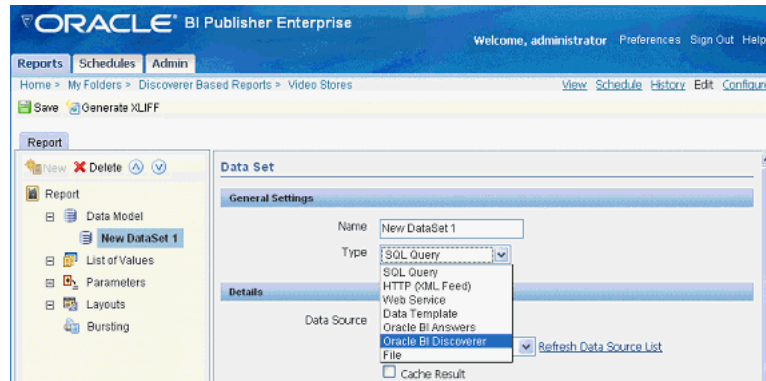
4. Click the **Edit** link under the report name.



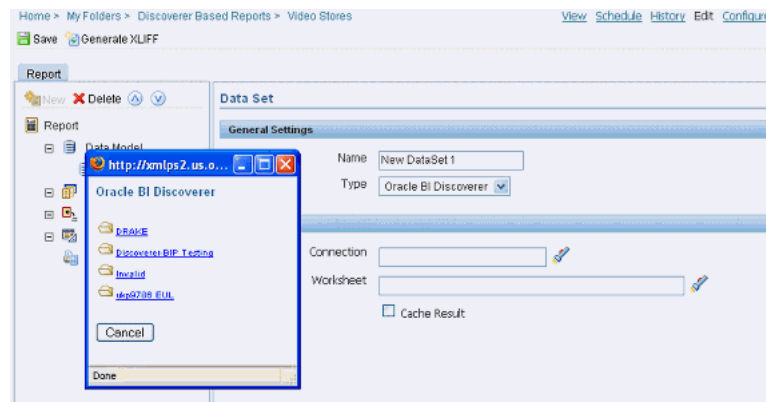
5. Select **Data Model** from the Report tree and click the **New** button at the top of the tree to create a new data set.



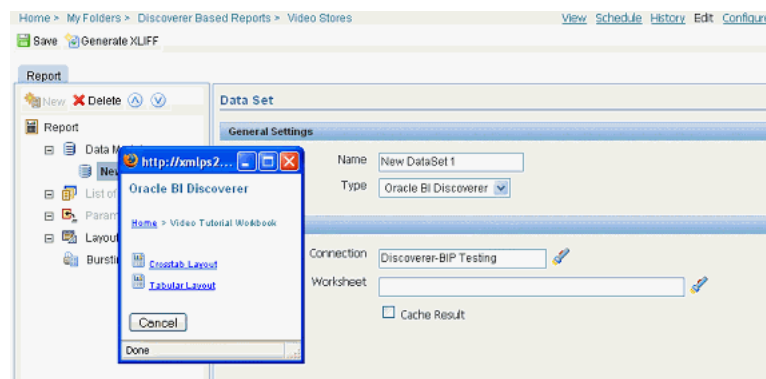
- Under the General Settings section of the data set, optionally name the data set and select **Oracle BI Discoverer** in the **Type** field.



- Under the Details section of the data set, click the flashlight icon to the right of the **Connection** field. Navigate to and select the Discoverer connection that owns the worksheet to use in the report.



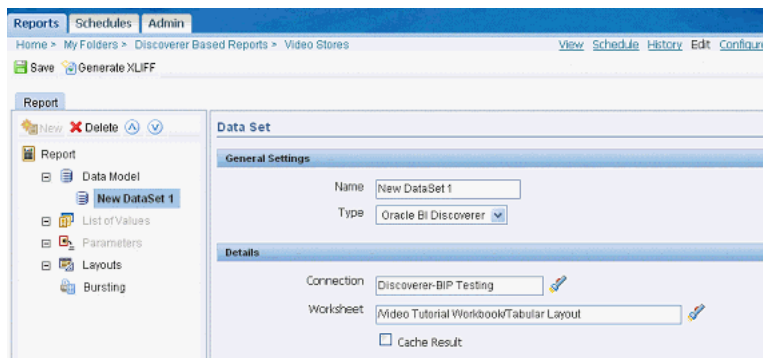
- After you have selected a connection, click the flashlight icon to the right of the **Worksheet** field. Navigate to and select the workbook and then the worksheet to use in the report.



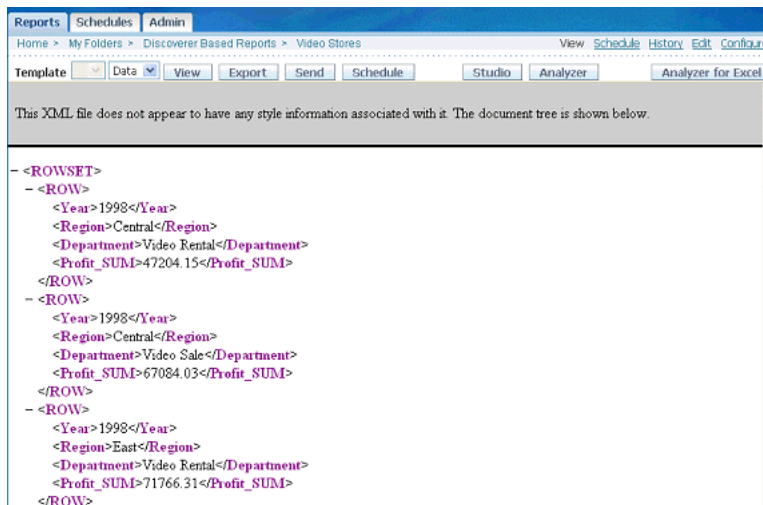
The fully qualified path to the worksheet is entered in the **Worksheet** field.

Regardless of whether the Discoverer worksheet is a tabular or crosstab layout, the Discoverer Web service returns the data to Oracle BI Publisher as flat tabular data. This is by design so that you can take full advantage of the layout capabilities in BI Publisher. You can lay out this data as a crosstab in BI Publisher.

9. Save the report by clicking the **Save** icon on the upper-left side of the page below the navigation links to the report.



10. When the report is saved, click the **View** link in the upper-right corner of the screen to the right of the navigation links to see that the Discoverer worksheet that you selected returns data to the report. If the worksheet contains parameters, then you might need to select values other than the default ones and click **View** to get the worksheet to return data.



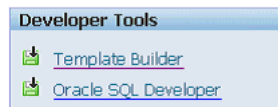
You have successfully created a report with a data model based on a Discoverer worksheet.

At this point, you can continue by creating a layout template by using either the online method or offline method. Continue with the section entitled [Section 18.2.3.1, "About Creating Layout Templates"](#).

18.2.3 Creating a Report Indirectly Through Oracle BI Publisher Template Builder

You can also create a report using Oracle BI Publisher Template Builder, an Add-in to Microsoft Office Word.

If you have not installed Oracle BI Publisher Template Builder, then download it by clicking the **Template Builder** link in the Developer Tools section on the left side of any page, visible in the BI Publisher Web interface when viewing any folder below Home in the report repository.

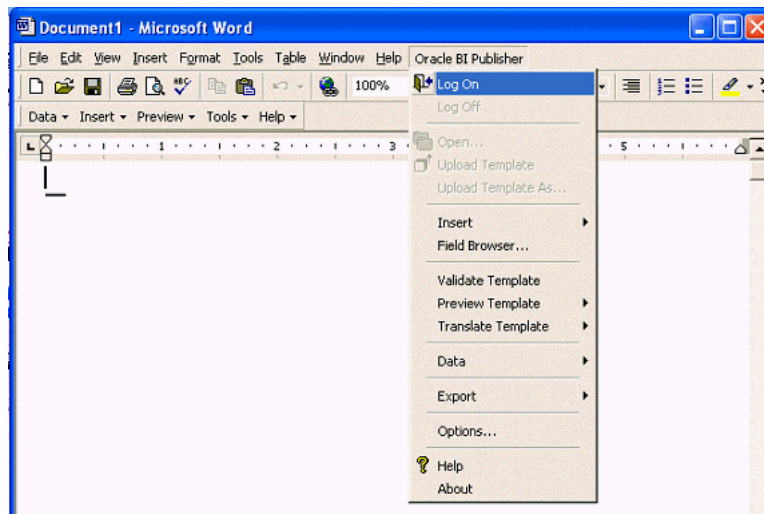


Then run the BIPublisherDesktop.exe that is downloaded. This installs Oracle BI Publisher Template Builder.

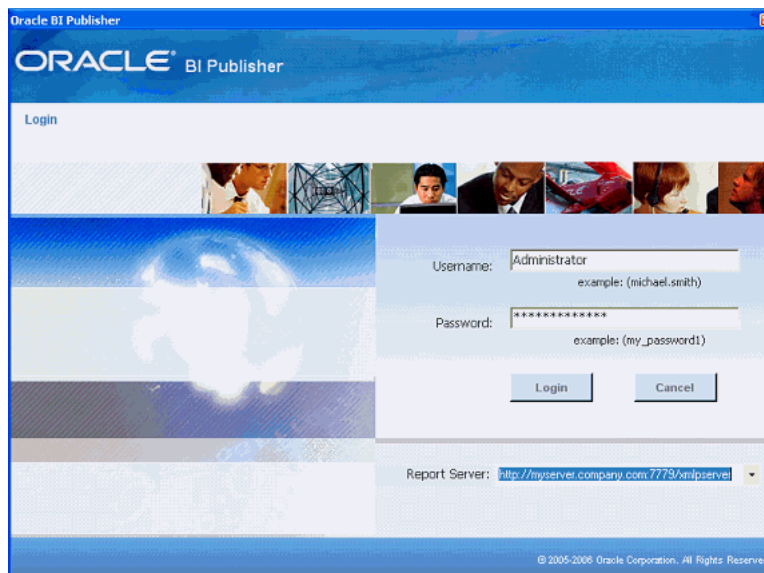
Note that you should exit all Microsoft Office products before installing and that you might also be prompted to download and install Microsoft .Net 2.0 if it was not previously installed.

After the installation is finished, continue with the following steps.

1. Start Microsoft Word. Connect to the Oracle BI Publisher server by selecting **Log On** from the Oracle BI Publisher menu.

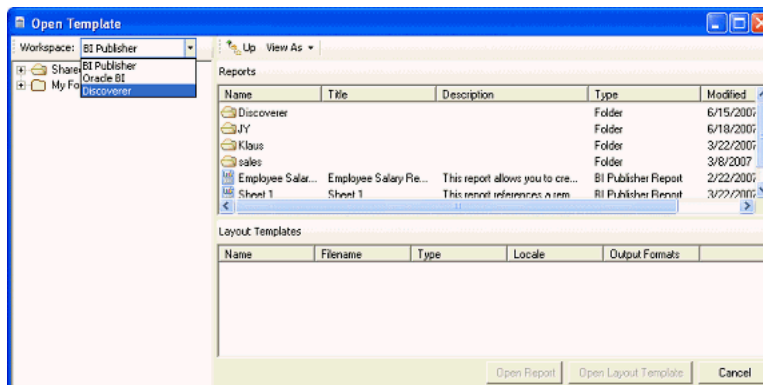


2. In the Oracle BI Publisher Login dialog, supply your Oracle BI Publisher user name and password. Note that, if this is the first time that you are logging into the Oracle BI Publisher server, then you must supply the report server URL (for example, myserver.company.com:7779/xmlpserver).

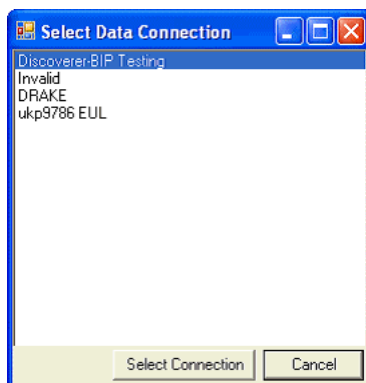


After the first log in, you can select the Report Server from the drop-down list.

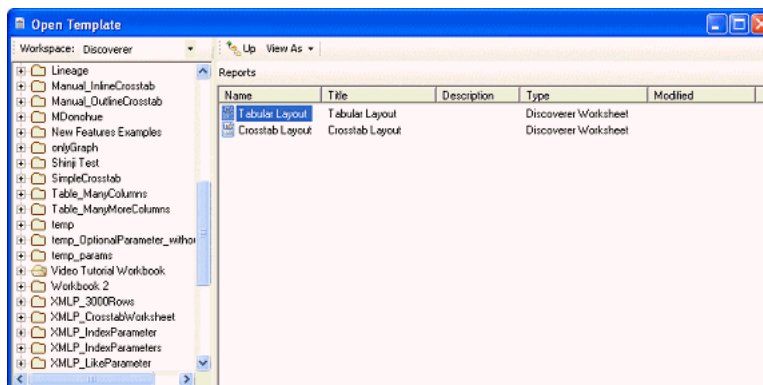
- When connected, you see the Open Template dialog. Select **Discoverer** from the Workspace list.



- Complete the Select Data Connection dialog.



- When you have selected a Discoverer connection, the Open Template dialog displays a list of workbooks on the left. As you select workbooks, the worksheets display in the right pane. Navigate through the workbooks and select a worksheet by double-clicking the worksheet name.



- You are prompted to save the worksheet as an Oracle BI Publisher report. You can navigate through the Oracle BI Publisher report repository and select the folder where the report is created.



A report is created that has a name of the form `workbook.worksheet` and it uses that worksheet as the data set. You cannot name or rename the report from this dialog or from the Template Builder. If a report exists with that name, then a new report is not created and the existing report remains as it is. If you want to rename the report, then use the Web interface to the Oracle BI Publisher report server.

At this point you can proceed with creating a layout template by following the online method described below. Continue with the section entitled [Section 18.2.3.1, "About Creating Layout Templates"](#).

18.2.3.1 About Creating Layout Templates

BI Publisher provides several types of layout templates. The most common are RTF layout templates created using the Template Builder. You can also create PDF templates in Adobe Acrobat, eText templates, and Adobe Flex templates. This document briefly describes only the process of creating an RTF layout template. See *Oracle Business Intelligence Publisher Report Designer's Guide* for more details on creating other types of templates and for more depth on RTF templates.

Note that even though a layout template created with the Template Builder is in RTF format, at run time the layout template is converted to Extensible Stylesheet Language Formatting Objects (XSL-FO) and the report output format can be PDF, HTML, Excel, PowerPoint, and RTF.

There are two methods you can use to create an RTF layout template for a report:

- **Offline** - meaning not connected to the Oracle BI Publisher report server. Using the offline method, you generate sample XML data for the report. This method has the advantage that you place no load on the report server or underlying data source while creating the layout. It is also the preferred method if the report returns a large amount of data or has parameters that must be set to limit the amount of data returned or to ensure that some data is returned.
- **Online** - meaning connected to the Oracle BI Publisher report server. The online method enables you to connect directly to the report server to download report data. This method is suitable if the report has been set up to provide a reasonable amount of data. This method also can be seamlessly integrated with the indirect method of creating a Discoverer report as described above.

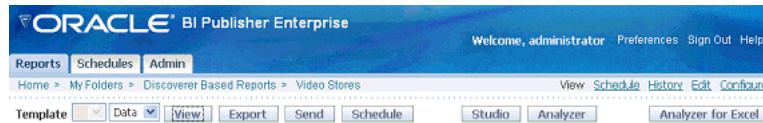
18.2.3.2 Creating a Layout Template Offline

1. Generate sample XML data.

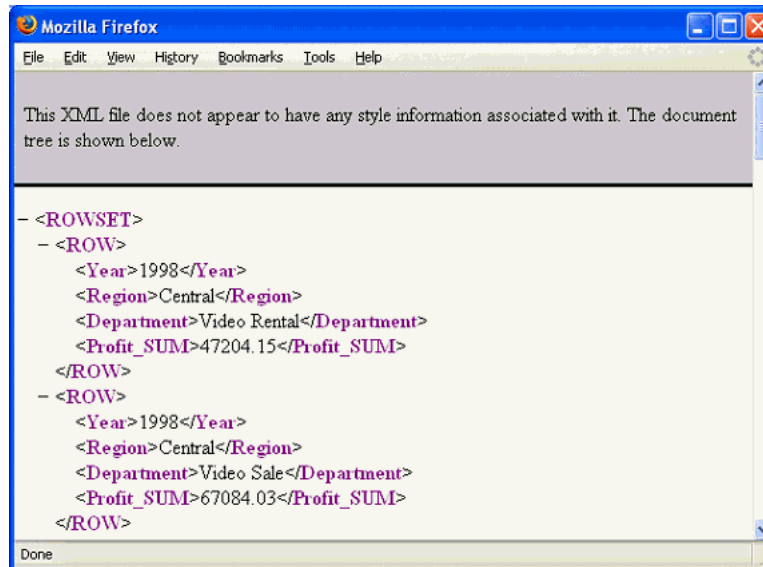
From the Oracle BI Publisher Web interface, navigate to the report to create a layout against, and select **View** to see the report output.

If there are parameters that can be set to either reduce the amount of data or to ensure that some data is returned, then set the parameter values and click the **View** button.

Then select Data from the menu to the left of the **View** button and click the **Export** button.



A new browser or tab opens that contains the exported XML data.

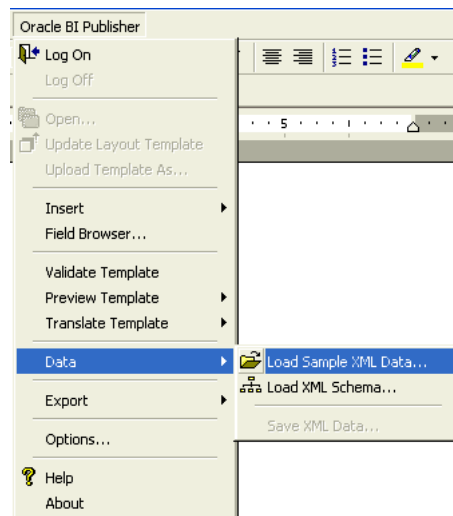


Using the Web browser's File menu, save the file with an XML extension.

If the size of the XML file is still excessive, then open it in a text editor and remove data from the file to get it to a smaller size. Ensure that you preserve the tag structure in the file. Ensure that you have enough data so that previewing the report shows how the actual report is generated. For example, if you plan to group data by year, then ensure that you include some data from at least two years, so that you can confirm that the grouping displays as you want.

2. Upload the sample XML data file to the Template Builder.

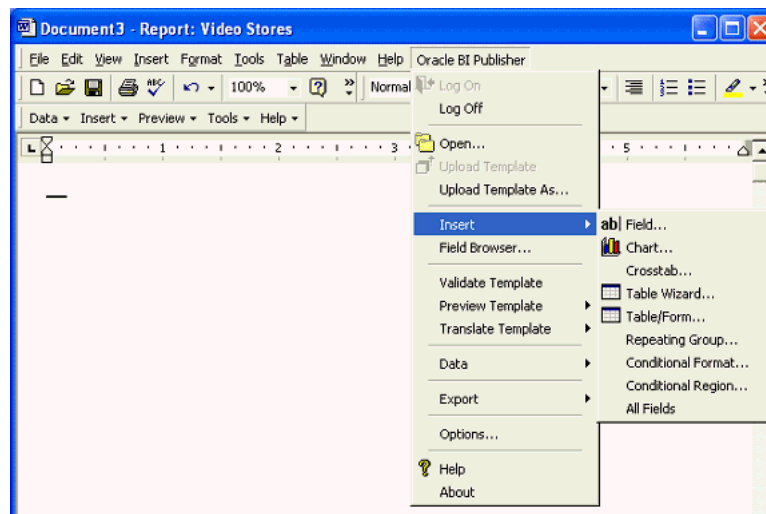
After you generate the sample XML data file, move it to the computer on which you run the Template Builder. From the Oracle BI Publisher menu or the Template Builder toolbar, load the sample XML data.



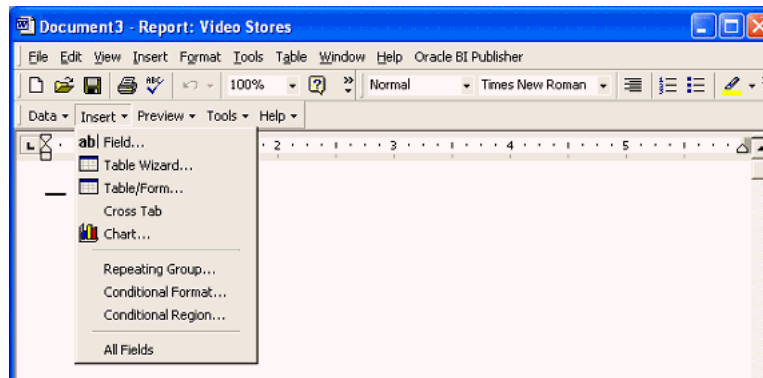
3. Create a layout template.

Now that you have uploaded the sample XML data to the Template Builder, you can use the Template Builder Insert menu and toolbar to launch wizards and dialogs that help you create the layout template. You can also insert any boilerplate text and you can format fields and tables using standard Microsoft Office Word layout and formatting functionality.

The following figure shows the Template Builder Insert Menu:



The following figure shows the Template Builder Insert Toolbar:



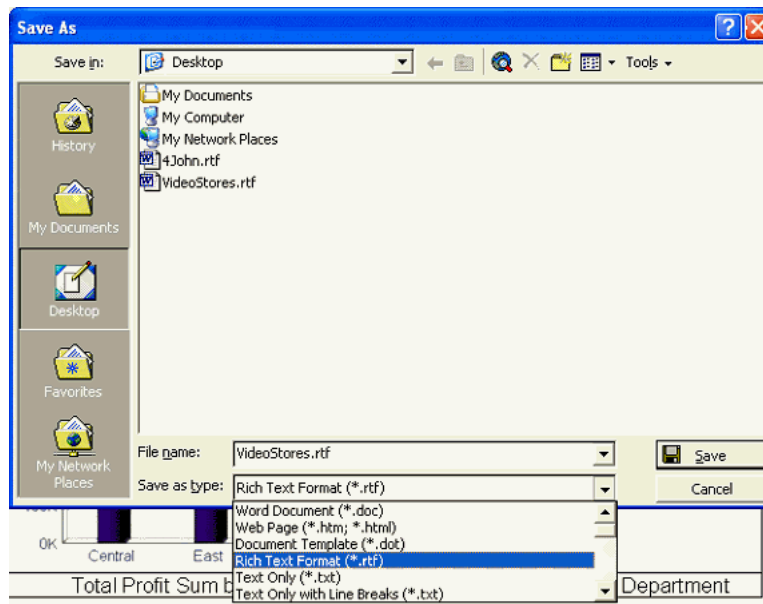
You can also manually include data field tags and other markup in the template using BI Publisher's simplified tags for XSL expressions. These tags associate the XML report data to the report layout. If you are familiar with XSL and prefer not to use the simplified tags, then Oracle BI Publisher also supports the use of pure XSL elements in the template.

See the section "Creating an RTF Template" in *Oracle Business Intelligence Publisher Report Designer's Guide* for more details, available from the Documentation pages on the Oracle Technology Network (OTN) or directly at:

http://download.oracle.com/docs/cd/E10091_01/doc/bip.1013/b40017/T421739T481157.htm

4. Save the layout template as an RTF file.

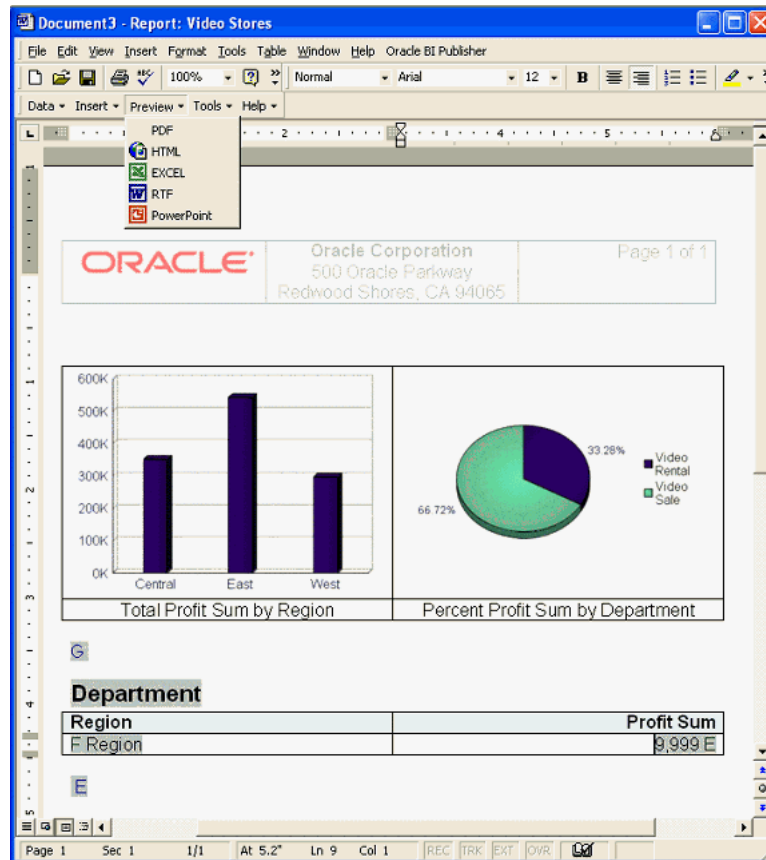
From the File menu, select **Save As**. Ensure that you set the type to Rich Text Format (*.rtf).



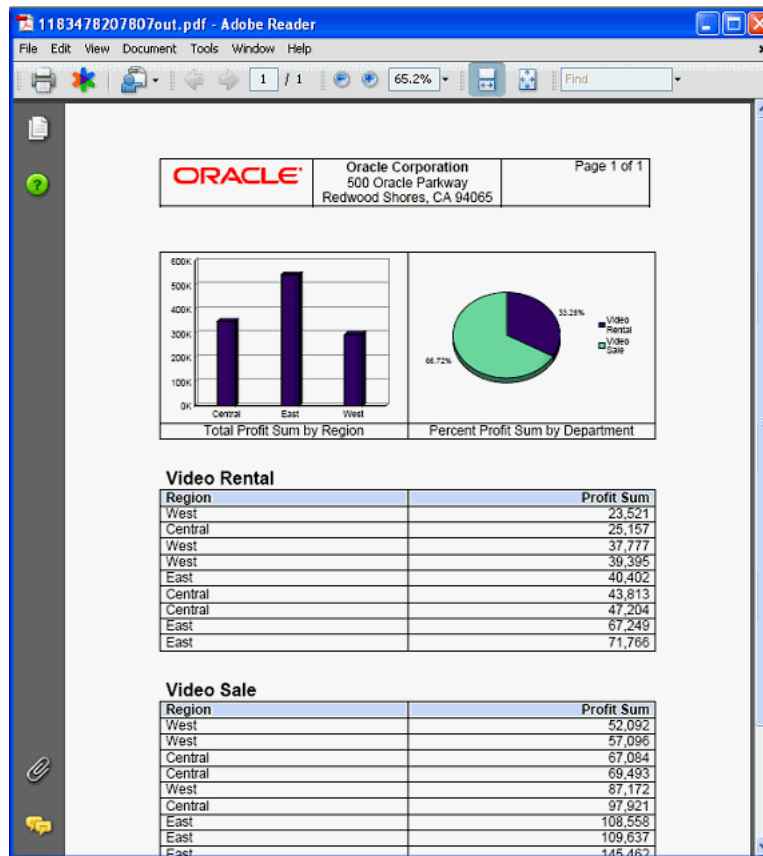
5. Preview the report output.

After you have created and formatted the layout, you can preview it in the desired output format from the Template Builder to confirm that you have created the layout exactly as you want. Previewing generates the output document, merging the layout template that you just created with the data that is loaded in the Template Builder.

Select the desired output format from the Preview menu item on the Template Builder toolbar or from the Oracle BI Publisher menu.



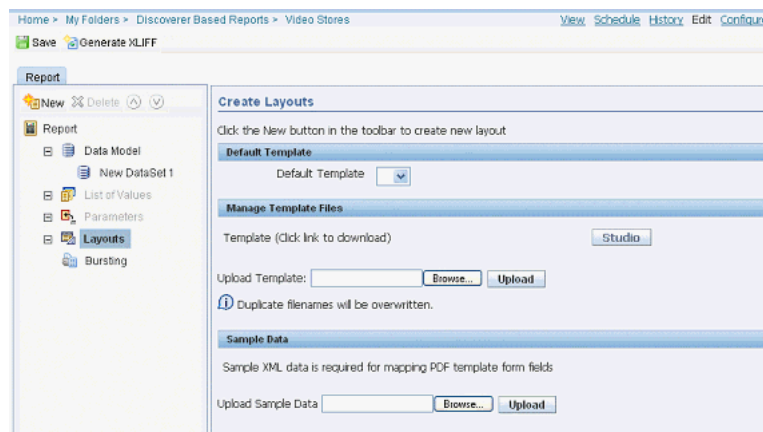
You see the generated document in the application suitable for displaying the requested output format. The following figure shows an example of previewing PDF output in Adobe Reader.



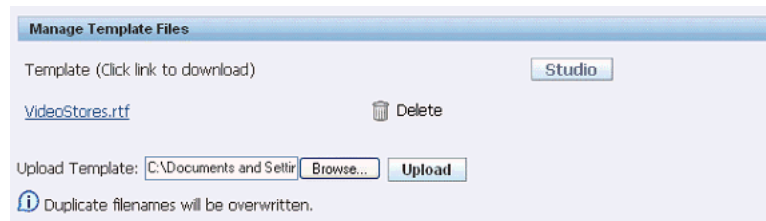
6. Upload the RTF template to the report.

When you have made the desired changes to the RTF template, save it to the file system. Return to the Oracle BI Publisher Web interface and navigate to the report that is associated with this layout template.

From within the report edit page of the Oracle BI Publisher web interface, select **Layouts** from the Report tree. In the Manage Template Files section of the Create Layouts frame, click the **Browse** button to select the RTF template. When the RTF template is selected and the **Upload Template** field is populated with the full path to the RTF template, click the **Upload** button.

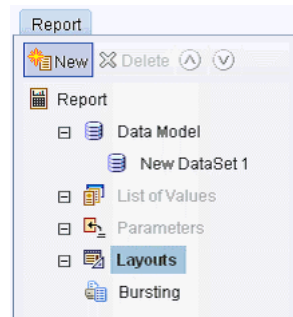


You then see the uploaded template in the Manage Template Files section of the page.

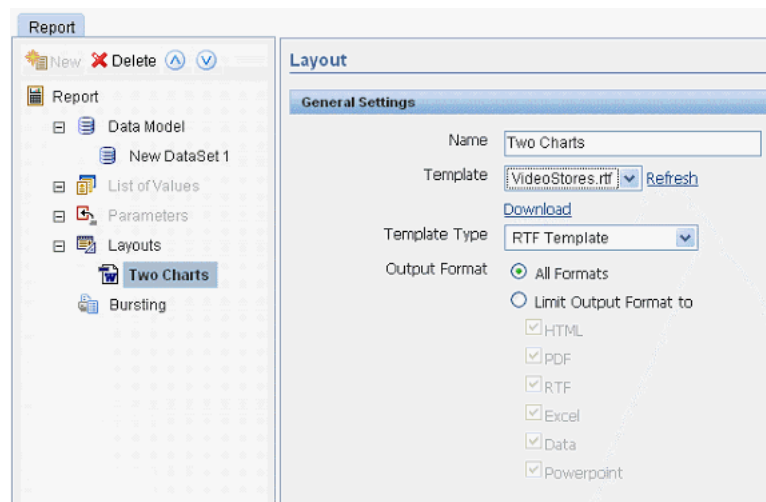


7. Create a named layout and associate it with the template file.

From the report tree, with Layouts still selected, click the **New** button at the top of the report tree to create a new layout.

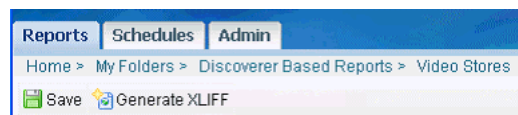


Provide a name for the layout and associate it with the template file that you uploaded. You can also optionally determine what output formats are available for this layout.



8. Save the report.

Ensure that you save the changes that you have made to the report by clicking the **Save** button below the path to the report.



9. Continue with [Section 18.2.3.4, "Viewing and Scheduling Reports"](#).

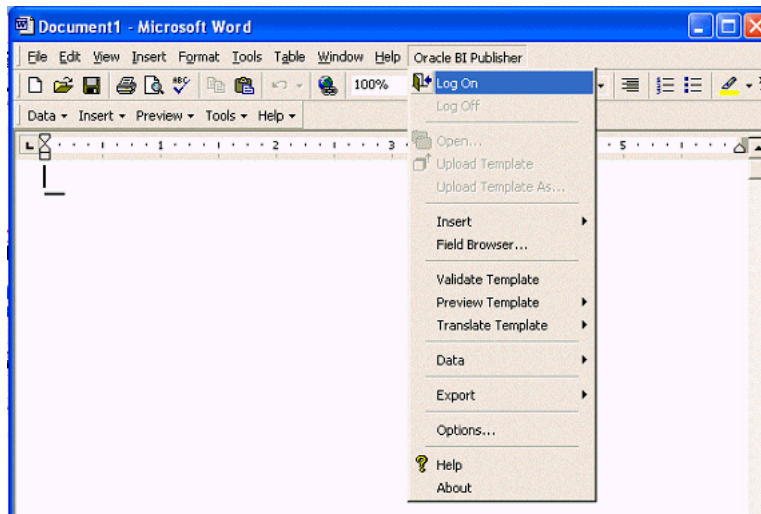
18.2.3.3 Creating a Layout Template Online

1. Connect to the Oracle BI Publisher report server through the Template Builder.

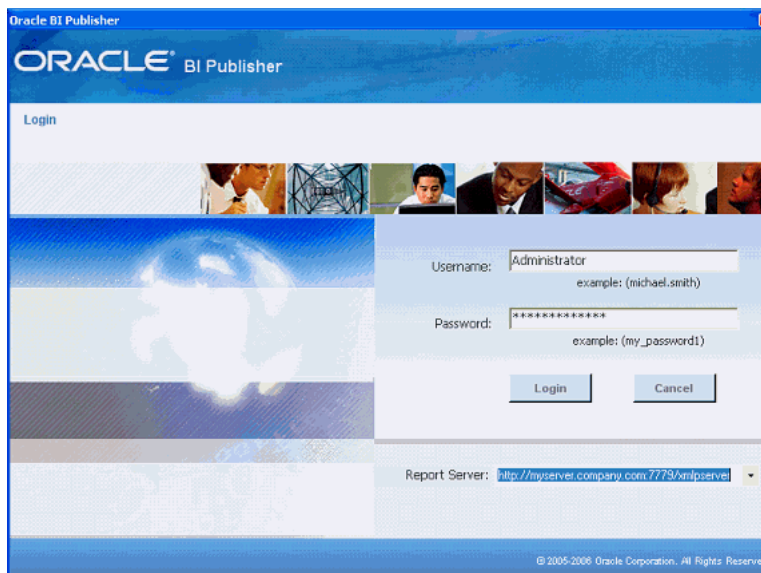
Note: If you have followed the method described above to create a report indirectly through the Oracle BI Publisher Template Builder, then you are logged on to the Oracle BI Publisher Server and the report is open. Skip ahead to Step 3 [Create a layout template](#).

If you have not previously logged on to the Oracle BI Publisher server, then do so by following these steps:

- a. Start Microsoft Word. Connect to the Oracle BI Publisher server by selecting **Log On** from the Oracle BI Publisher menu.



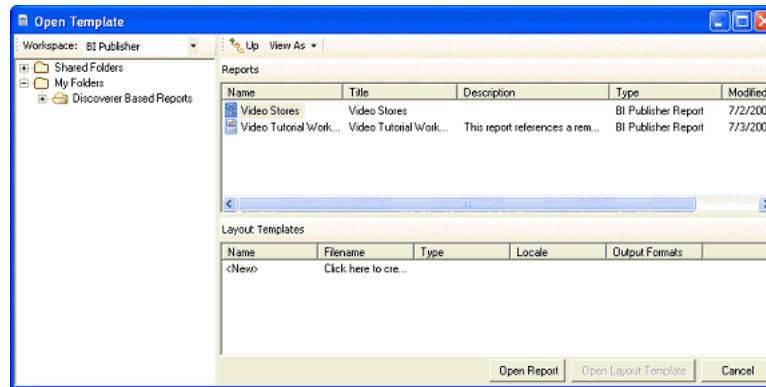
- b. In the Oracle BI Publisher Login dialog, supply your Oracle BI Publisher user name and password. Note that if this is the first time that you are logging into the Oracle BI Publisher server, then you must supply the report server URL (for example, myserver.company.com:7779/xmlpserver).



2. Create or open the desired report.

The method for creating a report indirectly through the Template Builder is described above in the section entitled [Section 18.2.3, "Creating a Report Indirectly Through Oracle BI Publisher Template Builder"](#).

To open an existing report, in the Open Template dialog, select BI Publisher as the Workspace. Navigate through the folder tree until the appropriate report is displayed in the list of reports.



Select the report name and click **Open Report**. Alternatively, select the report and then select <New> or any one of the existing layout templates listed in the Layout Templates section and click **Open Layout Template**.

When you have opened an existing report or layout template or created a new report or new layout template, the data model definition and fields and data are loaded into the Template Builder and you can start to create the layout template.

3. Create a layout template.

Now that you have opened or created a report based on a Discoverer worksheet, you can use the Template Builder Insert menu and toolbar to build the layout template.

See Step 3 in [Section 18.2.3.2, "Creating a Layout Template Offline"](#) for guidance on creating a layout template.

4. Save the layout template as an RTF file.

From the File menu, select **Save As**. Ensure that you set the type to Rich Text Format (*.rtf).

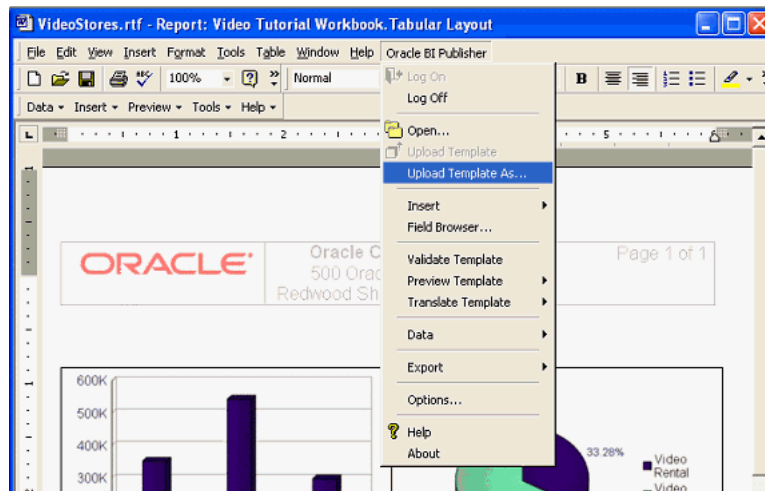
See Step 4 in [Section 18.2.3.2, "Creating a Layout Template Offline"](#).

5. Preview the report output.

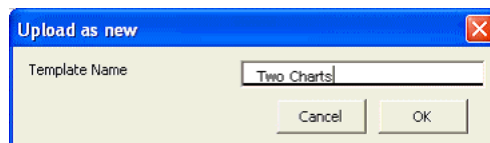
See Step 5 in [Section 18.2.3.2, "Creating a Layout Template Offline"](#).

6. Upload the RTF layout template to the Oracle BI Publisher server.

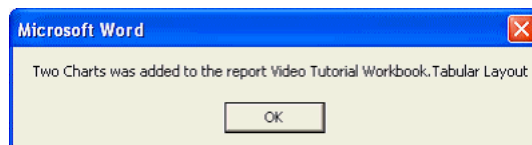
When you have completed the layout template, upload it to the BI server. Select **Upload Template As** from the Oracle BI Publisher menu.



If this is a new template, then you are prompted for a template name. Enter the template name and click **OK**.



When the template is uploaded, you see a notification dialog. Click **OK**.



The template has been added to the report.

7. Continue with [Section 18.2.3.4, "Viewing and Scheduling Reports"](#).

18.2.3.4 Viewing and Scheduling Reports

See "Viewing and Scheduling Reports" in *Oracle Business Intelligence Publisher Report Designer's Guide* available from the Documentation pages on the Oracle Technology Network (OTN) or directly at:

http://download.oracle.com/docs/cd/E10091_01/doc/bip.1013/b40017/T421739T421742.htm

18.2.4 Prerequisites, Limitations, and Known Issues

This section describes the prerequisites, limitation, and known issues in this release.

18.2.4.1 Prerequisites

The following are prerequisites for this release:

- Oracle BI Publisher 10.1.3.3
- Oracle BI Discoverer 10.1.2.3 or Discoverer 10.1.2.2 with a Discoverer one-off patch that includes the Discoverer Web Service
- Discoverer configured to use Oracle Application Server Identity Management

18.2.4.2 Limitations

The following are limitations with this release:

- Only one Discoverer Web Service can be configured per Oracle BI Publisher server or cluster.
- An Oracle BI Publisher report can contain only one data set based on a Discoverer worksheet.
- The Discoverer Web Service does not support Oracle Applications Single Sign On (SSO), Oracle Applications ICX and non-SSO modes.

18.2.4.3 Known Issues

The following are known issues in this release:

Issue: Unable to enter parameter values directly for parameters that have associated list of values.

Description: For Discoverer worksheets that have a parameter based on a List of Values (LOV) or an underlying item has an associated LOV, BI Publisher only allows parameter value selection from the LOV. Users cannot directly enter parameter values as they can in Discoverer.

Workaround: If users want to allow other users to enter parameter values directly, then they must remove the associated LOV in the Discoverer worksheet or item.

Issue: BI Publisher cannot communicate with the Discoverer Web Service when Discoverer is configured with Secure Socket Layer (SSL) (that is, uses https://)

Description: You might be unable to make any calls to the Discoverer Web Service. You might see errors such as "SSL received a record that exceeded the maximum permissible length."

Workaround: When using a Web service that is protected through SSL, you might need to export the certificate from the Web server that is hosting the Web service and import it into the Java keystore on the computer that is running BI Publisher. This step should not be required if the server certificate is linked to some certificate authority (such as Verisign); the default Java keystore can handle that. But if the Web service server is using some self-generated certificate (such as in a testing environment), then you must perform the export and import.

Issue: Discoverer server CPU reaches 100%.

Description: If a report that is based on a Discoverer worksheet takes several minutes to return data and a user becomes impatient and repeatedly clicks the View or Export buttons, then this can overwhelm the Discoverer Server with requests and it might stop responding.

Workaround: There is no workaround for this issue except to not repeatedly click the **View** or **Export** buttons while waiting for a report to return data.

Issue: BI Publisher does not return values for a Discoverer worksheet parameter's list of values past the 1000th value.

Description: Oracle BI Publisher limits the number of values in a list of values to 1000.

Workaround: Administrators can change this limit by setting the property DISCO_LOV_LIMIT in the xmlp-server-config.xml file. Example:

```
<property name=" DISCO_LOV_LIMIT " value="2000"/>
```

Note that the larger the list of values, the slower it renders.

Issue: See the following error when trying to open a report in the Template Builder:
"Error occurred. Please check the settings and try again."

Description: You might be loading a report that returns too much data for the Template Builder to handle.

Workaround: Use the method described above for creating a layout template offline. Ensure that you reduce the amount of data in the generated XML file.

Performance Improvements for Oracle BI Publisher in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

Release 10.1.3.3 has added performance improvements to BI Publisher's fast Formatting Objects Processor (FOP) and Bursting engine. If you run BI Publisher on a multiprocessor computer or on multicore processors, then you can take better advantage of the hardware.

To enable these performance improvements, display the Runtime Configuration page on the Admin page or follow the Configure link for a specific report.

This chapter includes the following sections:

- [Section 19.1, "Multithreading for Bursting"](#)
- [Section 19.2, "Multithreading for Document Generation"](#)

19.1 Multithreading for Bursting

If you have a multiprocessor computer or even a computer with a dual core single processor, then you might be able to achieve even higher bursting throughput using the new multithreading functionality for Bursting.

Under the Bursting section on the Runtime Configuration page, you set two properties. Set "Enable multithreading" to True and set "Thread count" to a number greater than 1 and up to the number of processors or cores on that computer.

If the report delivery channel is File System, then there is no considerable performance gain using multithreading. For delivery destinations other than File delivery, you should notice the performance gain.

Due to other processes that might be running on the system, you might need to determine empirically what is the optimal setting for "Thread count." Try a series of tests by varying the setting "Thread count" to see what is optimal for your environment.

Note: Leave these settings at the default if your system does not have multicore processors or multiple processors. Setting "Enable multithreading" to True and "Thread count" to a number greater than the number of cores on the computer leads to higher CPU usage without any gain in performance.

19.2 Multithreading for Document Generation

If you have a multiprocessor computer or even a computer with a dual core single processor, then you might be able to achieve even faster document generation using the new multithreading functionality for FO Processing.

Under the FO Processor section of the properties, set "Enable multithreading" to True.

The amount of performance gain that you see by enabling this setting depends on how much the current system resources are utilized.

On a system that runs numerous users and relatively high CPU utilizations, you likely see only minor improvements after setting "Enable multithreading" to True.

If the system is used by only a few users, if reports are scheduled sequentially one at a time, or if the number of CPUs is more than the number of concurrent reports, then turning on multiple threads speeds report generation.

Note: Memory utilization is likely to increase when "Enable multithreading" is set to True.

These properties can be set at the server and report level. See "Setting Runtime Properties" in *Oracle Business Intelligence Publisher Report Designer's Guide* for more information on setting properties.

Note: If you run BI Publisher on a single-core, one processor computer, then leave these multithreading configuration settings at the default value of False.

Converting Reports from Oracle Reports to Oracle BI Publisher in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

Release 10.1.3.3 introduces the ability to convert reports from Oracle Reports to BI Publisher, as described in this chapter. This chapter includes the following sections:

- [Section 20.1, "Overview"](#)
- [Section 20.2, "Prerequisites"](#)
- [Section 20.3, "Conversion Utility"](#)
- [Section 20.4, "Upload PL/SQL Package to the Database"](#)
- [Section 20.5, "Putting Converted Reports into the BI Publisher Repository"](#)
- [Section 20.6, "Testing and Tweaking the Reports"](#)

20.1 Overview

BI Publisher provides a utility for converting reports from Oracle Reports to BI Publisher.

In Oracle Reports, the data model (SQL query or extraction logic) and layout are contained in a single file. In BI Publisher the data model and the layout are separate objects. The conversion utility therefore generates several files that comprise the report in BI Publisher. In most cases this includes a PL/SQL specification and body that you must create in the database that contains the data for the report. The utility also generates a report definition file and a layout template file that you upload to the BI Publisher repository.

When uploaded, test the report to ensure that the output is as expected and make any changes to the report as needed. Some reports contain structures that the utility cannot convert. These must be manually implemented in the converted reports.

The overall flow for the conversion process is as follows:

1. Run the conversion utility.
2. Load the PL/SQL package into the database.
3. Upload the report to the BI Publisher repository.

4. Test the report and check the conversion log files to identify any manual modifications needed to complete the conversion.

20.2 Prerequisites

This section describes the steps required before running the conversion utility.

20.2.1 Reports and Oracle Reports XML Format

During the conversion process, the source reports must eventually be in Oracle Reports XML format. Oracle Reports XML format is supported in Oracle Reports 9i and above.

If the source reports are not in Oracle Reports XML format, the conversion utility handles this conversion automatically. However, this requires Oracle Reports Designer 9i or later on the same computer where you perform the conversion so that the BI Publisher conversion utility can call the `rwconverter` executable to get the reports into Oracle Reports XML format.

To get Oracle Reports Designer, download Oracle Developer Suite 10g (10.1.2.0.2) from:

<http://www.oracle.com/technology/software/products/ids/index.html>

and install Reports Designer.

If the reports exist in Oracle Reports XML format, there is no need to install Oracle Reports.

20.2.2 Update CLASSPATH with Needed JAR Files

To run the conversion utilities, you need the following jar files in your classpath:

- Collections.zip -
- xmlparserv2-904.jar -
- xdoccore.jar
- aolj.jar

For customers using Oracle Enterprise Business Suite (EBS), these libraries are available under `JAVA_TOP`. If these libraries are not there, then the corresponding classes exist under the `JAVA_TOP`.

For customer using Oracle BI Publisher Enterprise (sometimes referred to as "standalone"), all these libraries are available under `WEB-INF\lib`. Example:

```
C:\Oracle\bi\oc4j_  
bi\j2ee\home\applications\xmlpserver\xmlpserver\WEB-INF\lib
```

20.2.3 JDK Version 1.1.8

The conversion utility requires JDK version 1.1.8 or later.

20.3 Conversion Utility

`BIPBatchConversion` is the utility to migrate reports from Oracle Reports to BI Publisher.

It takes the following parameters:

- **-source** — (required) Source directory for Oracle Reports files. All reports must be in the same format - either RDF or XML.
- **-target** — (required) Target directory to create BI Publisher report objects. This includes the BI Publisher Report file (.xdo), the layout template file (.rtf), the PL/SQL package, and log file.
- **-oraclehome** — (optional) If the reports are in Oracle Reports XML format do not specify this parameter. If the reports are not in Oracle Reports XML format, specify the Oracle home path where Oracle Report Designer (9i or later version) is installed. BIPBatchMigration assumes that rwconverter is contained in the bin directory beneath the Oracle home path.

BIPBatchConversion requires rwconverter from Oracle Reports to convert the report from RDF format to XML format.

- **-debug** — (optional) To run the utility in debug mode and write debug statements to the log file.

Command Line Usage:

```
java ... BIPBatchConversion [-debug] -source SourceDirectory -target
TargetDirectory [-oraclehome OracleHomePath]
```

Sample Commands to run BIPBatchConversion:

Example 20–1 Source Report in Oracle Reports RDF File

This example requires you to specify the -oraclehome path.

```
java.exe -classpath
D:\Jdev\project\xdocore.jar;d:\Jdev\project\collections.zip;d:\Jdev\project\ao1j.jar;
d:\Jdev\project\xmlparserv2-904.jar
oracle.apps.xdo.rdfparser.BIPBatchConversion -source d:\reports\pay -target
d:\reports\pay\output -oraclehome D:\oracle\BIToolsHome_1 -debug
```

Example 20–2 Source Report in Oracle Reports XML Format

Do not specify the -oraclehome parameter in this example.

```
java.exe -classpath
D:\Jdev\project\xdocore.jar;d:\Jdev\project\collections.zip;d:\Jdev\project\ao1j.jar;
d:\Jdev\project\xmlparserv2-904.jar
oracle.apps.xdo.rdfparser.BIPBatchConversion -source d:\reports\pay -target
d:\reports\pay\output -debug
```

See the Oracle BI Publisher 10.1.3.3 Java API reference (Javadoc) for more details on this utility and other related conversion APIs in the rdfparser package.

20.3.1 Output Files

The conversion utility generates the following output files for each report:

- Report definition file that includes the Data Model (for example: REPORT.xdo)
(Note: This file is not needed for E-Business Suite users; see the following note under DataTemplate.)
- DataTemplate (for example: REPORT_template.xml)

(Note: this is not required for Oracle BI Publisher Enterprise users because the data template is embedded in REPORT.xdo.)

- Default PL/SQL package specification (for example: REPORTS.pls)
- Default PL/SQL package body (for example: REPORTB.pls)
- RTF Layout Template (for example: REPORT.rtf)
- Log file (for example: REPORT.log)

For example, assume that you have a report called raxinv.rdf located in D:\reports\pay. Running this command:

```
java.exe -classpath
D:\Jdev\project\xdocore.jar;d:\Jdev\project\collections.zip;d:\Jdev\project\ao1j.jar;d:\Jdev\project\xmlparserv2-904.jar
oracle.apps.xdo.rdfparser.BIPBatchConversion -source d:\reports\pay -target
d:\reports\pay\output -oraclehome D:\oracle\BIToolsHome_1 -debug
```

Generates the following output files:

- PL/SQL Package Specification: C:\BIPublisher_reports\raxinv\raxinvS.pls
- PL/SQL Package Body: C:\BIPublisher_reports\raxinv\raxinvB.pls
- Report definition file: C:\BIPublisher_reports\raxinv\raxinv.xdo
- DataTemplate: C:\BIPublisher_reports\raxinv\raxinv_template.xml
- RTF Layout Template: C:\BIPublisher_reports\raxinv\raxinv.rtf
- Log file: C:\BIPublisher_reports\raxinv\raxinv.log

20.4 Upload PL/SQL Package to the Database

Many converted Oracle Reports generate a PL/SQL package specification file and a PL/SQL package body file as follows:

- <report_name>S.pls
- <report_name>B.pls

Run the PL/SQL package files against the Oracle Database as follows. These files create the PL/SQL package specification and body.

```
SQL> @C:\BIPublisher_reports\raxinv\raxinvS.pls
SQL> @C:\BIPublisher_reports\raxinv\raxinvB.pls
```

20.5 Putting Converted Reports into the BI Publisher Repository

Making your reports visible in the BI Publisher repository is a two-step process:

1. Copy the report folders into the repository.
2. Refresh the repository metadata.

If you have a file-based repository, then copy the report folder structure to the file system. For example, suppose the repository path was set to:

C:\oracle\bi\xmlp\XMLP

For the report in the previous example, simply copy the report directory and all its files to the desired folder structure in the existing report repository.

For example:

```
> copy C:\BIPublisher_reports\raxinv C:\oracle\bi\xmlp\XMLP\Shared
Folders\Converted reports
```

If you are using the XML DB functionality of the Oracle Database as the report repository, then there are several ways that you can upload content. One method is through WebDAV.

For options and details on how to load reports into the XML DB based repository, see the following topics in *Oracle XML DB Developer's Guide* (http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14259/toc.htm):

- "Loading XML Content into Oracle XML DB"
- "FTP, HTTP(S), and WebDAV Access to Repository Data"

When you have loaded report directories into the repository, log on to Oracle BI Publisher with administrator privileges and select "Refresh Metadata" from the System Maintenance Section of the Oracle BI Publisher Admin page. Now all reports should be available for testing and execution.

20.6 Testing and Tweaking the Reports

Now that you have successfully converted the reports, created the needed PL/SQL package, and moved the report definition and RTF template files into the BI Publisher repository, you should test the reports to ensure that they are providing the appropriate data and formatting.

Most converted reports run exactly as expected without further modification. More complex reports might require some additional modification to work as desired. Following are some common issues with converted reports.

20.6.1 Data Template and PL/SQL Files

Occasionally when converting a more complex Oracle Reports report, the Data Template or PL/SQL might contain minor errors and require manual correction.

The conversion utility moves all formula columns to the select clause of the SQL query in the data model. In most cases this does not cause a problem. However, if any of the arguments to the formula is a summary column, this does not work because the summary column is not calculated at the time the query is executed.

To correct this problem, remove this formula from the select clause and implement the formula as XSL in the layout template. Most of these formulas are used either for simple addition or summation or currency conversion, formatting, and rounding.

20.6.2 RTF Layout Template File

The majority of Oracle Reports reports use simple "if" formatting logic that is automatically converted to equivalent XSL-FO and inserted into form fields in the RTF layout template. However, there is no support for PL/SQL in RTF layout templates. The conversion utility does not convert any PL/SQL format trigger logic present in the report. Instead the conversion utility writes all the format trigger code to a log file. You must implement any corresponding PL/SQL logic as XSL code.

To aid in this process, the resulting RTF template contains form fields that hold the format trigger names that are called; these fields are highlighted in red. You can then refer to the log to find the actual PL/SQL code used in the original Oracle Report. You must rewrite these PL/SQL triggers as XSL-FO.

For more information, see the "Supported XSL-FO Elements" section of *Oracle Business Intelligence Publisher Report Designer's Guide*
(http://download.oracle.com/docs/cd/E10091_01/doc/bip.1013/b40017/toc.htm)

Support for Digital Signature in PDF Documents in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

Release 10.1.3.3 introduces BI Publisher's support for Digital Signature in PDF Documents, as described in this chapter. This chapter includes the following sections:

- [Section 21.1, "Introduction"](#)
- [Section 21.2, "Prerequisites"](#)
- [Section 21.3, "Limitations"](#)
- [Section 21.4, "Integration with BI Publisher"](#)

21.1 Introduction

You can add a digital signature to PDF output documents. Digital signatures enable you to verify the authenticity of the documents that you send and receive.

BI Publisher can access a digital ID file from a central, secure location and at run time, sign the PDF output with the digital ID. The digital signature verifies the signer's identity and ensures that the document has not been altered after it was signed.

For additional information on digital signatures, see the following sources:

- Digital ID Introduction by Verisign
 - <http://www.verisign.com/support/tlc/per/whitepaper.htm>
 - <http://www.verisign.com/stellent/groups/public/documents/guides/005326.pdf>
- Digital Signature by Adobe
 - <http://www.adobe.com/security/digsig.html>
- Digital Signatures in PDF and Acrobat from the following location:
 - <http://www.acrobatusers.com>

21.2 Prerequisites

Before you can implement digital signatures with BI Publisher output documents, you need the following:

- A digital ID obtained from a public certificate authority or from a private/internal certificate authority (if for internal use only). You must copy the digital ID file to a secure location of the file system on the server that is accessible by the BI Publisher server.

21.3 Limitations

Use of digital signatures with BI Publisher output documents has the following limitations:

- Only a single digital ID can be registered with BI Publisher in this release. Future releases will support multiple digital IDs.
- Only reports submitted through BI Publisher's Schedule Report interface can include the digital signature.
- The digital signature is enabled at the report level; therefore, multiple templates assigned to the same report share the digital signature properties.

21.4 Integration with BI Publisher

The following steps summarize the tasks required to set up and use digital signature with BI Publisher:

1. Register the digital ID in the BI Publisher Admin interface.
2. Specify the Roles that are authorized to sign documents.
3. (Optional for PDF templates) Add a signature field to the PDF template in which to place the digital signature at run time.
4. Enable Digital Signature for the report in the Runtime Configuration properties and specify the position to place the digital signature on the completed document. This can be a signature field (for PDF templates), general location (top left, top center, or top right), or you can specify x and y coordinates.
5. Log in to BI Publisher as a user with an authorized role and submit the report through the BI Publisher scheduler, choosing PDF output. When the report completes, it is signed with your digital ID in the specified location of the document.

21.4.1 Digital Signature ID Registration and Allowed Role Assignment

In this release, BI Publisher supports the identification of a single digital ID file only. Register the digital ID in the BI Publisher Admin page as follows:

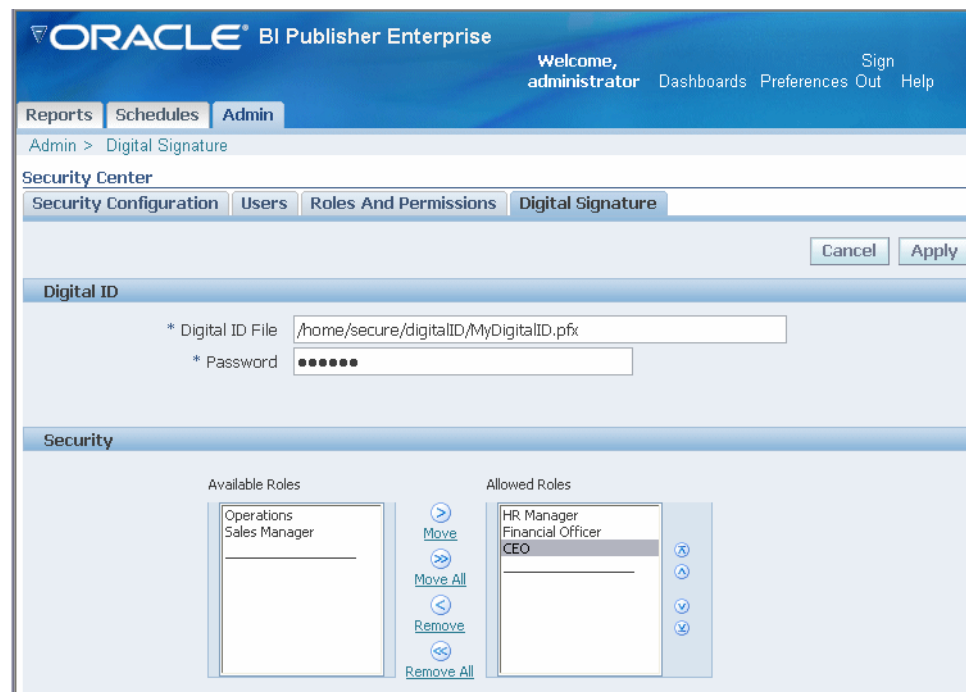
1. Log in to BI Publisher with Administrator credentials.
2. Select the Admin tab, and then from the Security Center menu select Digital Signature.

The following screenshot shows the new Digital Signature entry available from the Admin tab.



3. On the Digital Signature sub-tab, enter the file path to the digital ID file and enter the password for the digital ID.
4. Enable the Roles that have the authority to sign documents with this digital ID. Use the shuttle buttons to move Available Roles to the Allowed Roles list.
5. Click Apply.

The following screenshot shows the Digital Signature sub-tab:



21.4.2 Specify the Signature Display Field or Location

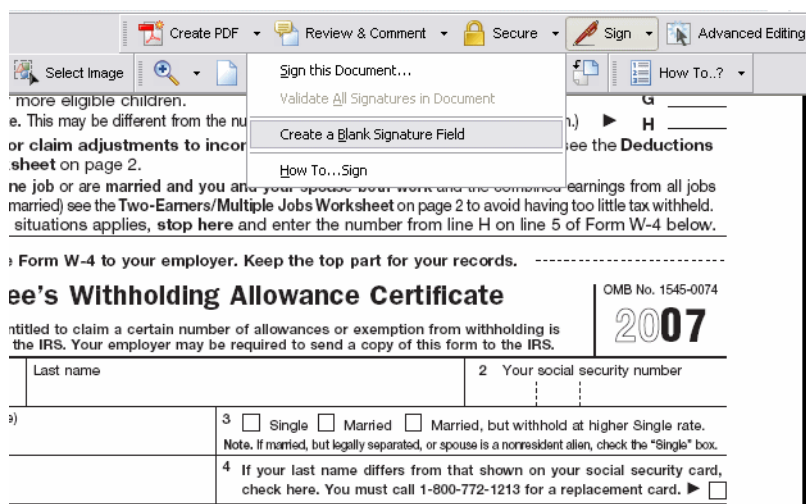
You must specify the location for the digital signature to display in the completed document. If the template is PDF, then you can either specify a template field in which to put the digital signature or you can just specify a location on the document using Runtime Configuration properties. If the template is RTF, then you can only specify a location; you cannot specify a field.

21.4.2.1 Designating a Field in a PDF Template for the Digital Signature

If the template is a PDF, then you can designate an existing field in which to place the digital signature or insert a new signature field on the form.

To add a signature field:

1. Open the template in Adobe Acrobat Professional.
2. Select Sign from the toolbar and select Create a Blank Signature Field. The following figure displays this selection from the Adobe Acrobat Professional toolbar.



3. Draw the signature field in the desired location on the template. When you are finished drawing (that is, when you release the mouse button), the Digital Signature Properties dialog is displayed.
4. Assign a name to your signature field. The following figure shows an inserted digital signature field called "My_Signature."

For accuracy, complete all worksheets that apply.

- If you plan to **itemize or claim adjustments to income** and want to reduce your withholding, see the **D and Adjustments Worksheet** on page 2.
- If you have more than one job or are married and you and your spouse both work and the combined earnings exceed \$40,000 (\$25,000 if married), see the **Multiple Jobs Worksheet** on page 2.
- If neither of the above situations apply, you can skip this worksheet.

W-4 Employee's
Department of the Treasury
Internal Revenue Service

1 Type or print your first name and middle initial. Last

Home address (number and street or rural route)

City or town, state, and ZIP code

5 Total number of allowances you are claiming

6 Additional amount, if any, you want withheld

7 I claim exemption from withholding for 2007, if:

- Last year I had a right to a refund of all federal income tax withheld.
- This year I expect a refund of all federal income tax withheld because I expect to have no tax liability.

If you meet both conditions, write "Exempt" here

Under penalties of perjury, I declare that I have examined this certificate and to the best of my knowledge and belief, it is true, correct, and complete.

Employee's signature
(Form is not valid unless you sign it.)

My_Signature

Date

8 Employer's name and address (Employer: Complete lines 8 and 10 only if sending to the IRS.) 9 Office code (optional) 10 Employer identification

Digital Signature Properties

General Appearance Actions Signed

Name My_Signature

Tooltip

Common Properties

Form Field: Visible Read Only

Orientation: 0 degrees Required

Locked Close

5. Save the template.

Now perform the following task to configure BI Publisher to insert the digital signature to this field.

To upload the template and configure the report:

1. Log in to BI Publisher with Developer or Administrator credentials.
2. Upload the template to the appropriate report definition.
 - a. Navigate to the report.
 - b. Select the Edit link.
 - c. From the Report definition pane select Layouts.
 - d. Select Browse to locate the template, then select Upload.
 - e. After the template uploads successfully, with Layouts still selected on the Report definition pane, select New to add a layout.
 - f. In the General Settings of the Layout definition, enter a Name for the layout, then select the uploaded template from the list. Select the appropriate Template Type and Output Formats.
 - g. Click Save.
3. Select the Configure link for the report.
4. Scroll down to the PDF Digital Signature group of properties.
5. Set Enable Digital Signature to True.
6. Enter the field name from the PDF template for the property Existing signature field name.

The following screenshot shows the "My_Signature" field name entered into the properties field.

	Report Value	Server Value
Properties		
Bursting		
Enable multithreading	<input type="checkbox"/>	False
Thread count	<input type="text" value="2"/>	2
PDF Output		
Compress PDF output	<input type="checkbox"/>	True
Hide PDF viewer's menu bars	<input type="checkbox"/>	False
PDF Digital Signature		
Enable Digital Signature	<input checked="" type="checkbox"/>	False
Existing signature field name	<input type="text" value="My_Signature"/>	
Signature field location	<input type="text"/>	
Signature field X coordinate	<input type="text"/>	0
Signature field Y coordinate	<input type="text"/>	0
Signature field width	<input type="text"/>	0
Signature field height	<input type="text"/>	0
RTF Output		

7. Select Apply.

Note that the run-time properties that you have just set are at the report level and not the template level. Therefore any templates associated with the report now include the digital signature as specified in the Runtime Configuration properties for the report. When an Existing signature field name is specified, the template must contain the field for the signature to be applied.

21.4.2.2 Specifying a Location in the Document to Insert the Digital Signature (for RTF and PDF Templates)

When you specify a location in the document to place the digital signature, you can either specify a general location (Top Left, Top Center, or Top Right) or you can specify x and y coordinates in the document. You can also specify the field height and width. This is done through properties on the Runtime Configuration page. Therefore you do not need to alter the template to include a digital signature.

1. Log in to BI Publisher with Developer or Administrator credentials.
2. Navigate to the report that allows authorized users to sign with the digital ID.
3. Select the Configure link for the report.
4. Scroll to the PDF Digital Signature group of properties.
5. Set Enable Digital Signature to True.
6. Specify the location in the document where you want the digital signature to display by setting the appropriate properties as follows (note that the signature is inserted on the first page of the document only):
 - Existing signature field name – does not apply to this method.
 - Signature field location – provides a list that contains the following values: Top Left, Top Center, Top Right. Choose one of these general locations and BI Publisher inserts the digital signature to the output document sized and

positioned appropriately. If you set this property, then do not enter X and Y coordinates or width and height properties.

- Signature field X coordinate – using the left edge of the document as the zero point of the X axis, enter the position in points for placing the digital signature from the left. For example, if you want the digital signature to be placed horizontally in the middle of an 8.5 inch by 11 inch document (that is, 612 points in width and 792 points in height), enter 306.
- Signature field Y coordinate – using the bottom edge of the document as the zero point of the Y axis, enter the position in points for placing the digital signature from the bottom. For example, if you want the digital signature to be placed vertically in the middle of an 8.5 inch by 11 inch document (that is, 612 points in width and 792 points in height), enter 396.
- Signature field width – enter in points the desired width of the inserted digital signature field. This applies only if you are setting the X and Y coordinates.
- Signature field height – enter in points the desired height of the inserted digital signature field. This applies only if you are setting the X and Y coordinates.

The following screenshot shows a report configured to place the digital signature at specific x and y coordinates in the document.

	Report Value	Server Value
<input checked="" type="checkbox"/> Properties		
<input checked="" type="checkbox"/> Bursting		
Enable multithreading	<input type="checkbox"/>	False
Thread count	<input type="text" value="2"/>	2
<input checked="" type="checkbox"/> PDF Output		
Compress PDF output	<input type="checkbox"/>	True
Hide PDF viewer's menu bars	<input type="checkbox"/>	False
<input checked="" type="checkbox"/> PDF Digital Signature		
Enable Digital Signature	True <input type="checkbox"/>	False
Existing signature field name	<input type="text"/>	
Signature field location	<input type="checkbox"/>	
Signature field X coordinate	<input type="text" value="306"/>	0
Signature field Y coordinate	<input type="text" value="700"/>	0
Signature field width	<input type="text" value="72"/>	0
Signature field height	<input type="text" value="36"/>	0
<input checked="" type="checkbox"/> RTF Output		

21.4.3 Running and Signing Reports with the Digital Signature

Users assigned a role with the digital signature privilege can attach the digital signature to their generated reports. The digital signature can only be inserted on scheduled reports.

1. Log in to BI Publisher as a user with a role granted digital signature privileges.
2. Navigate to the report that has been enabled for digital signature and select Schedule.
3. Complete the scheduler fields, selecting PDF output, and then submit the job.

4. The completed PDF displays the digital signature.

21.4.4 Additional PDF Security Features

BI Publisher also supports secured PDF to prevent unauthorized users from opening, printing, updating or cutting and pasting the PDF document contents. Digital signature is also supported for PDF reports configured with these security features enabled.

Users with the Developer or Administrator role can change PDF security settings for a report. To set PDF security properties, select the Configure link for the report. For more information about the PDF security properties, see the topic "Setting Run-time Properties" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

Support for PostScript Printers in Release 10.1.3.3

This chapter describes a feature of BI Publisher that was introduced in Release 10.1.3.3 and that applies to Releases 10.1.3.3 through 10.1.3.4. Information about this feature is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4.

Release 10.1.3.3 introduces BI Publisher's support for PostScript printers, as described in this chapter. This chapter includes the following sections:

- [Section 22.1, "Introduction"](#)
- [Section 22.2, "Setting Up the Print Server to Use the Converter"](#)
- [Section 22.3, "Limitations"](#)

22.1 Introduction

PDF is a popular output format for business reports and is printable from viewer software such as Adobe Reader. However, some reports require printing directly from the report server. For example, paychecks and invoices are usually printed as scheduled batch jobs. Some newer printers with PostScript Level 3 compliant Raster Image Processing can natively support PDF documents, but there are still many printers in business use that support only PostScript Level 2 that cannot print PDF documents directly.

BI Publisher provides native support for PostScript Level printers. Previously, to enable printing from the BI Publisher server to a PostScript printer, you configured a filter to invoke a third-party utility to perform the conversion. Now BI Publisher provides the utility to convert PDF to PostScript 2.0. If the printer supports PostScript 2.0, then you are no longer required to use a third-party utility to print directly from the server.

22.2 Setting Up the Print Server to Use the Converter

The conversion utility is provided as an option from the Filter list of the Print Server definition page:

1. Log in to Oracle BI Publisher as an Administrator.
2. From the Admin tab, under the Delivery heading, select Printer.
3. Select an existing print server, or to add a new one, click Add Server.
4. Under the General properties, select PDF to PostScript from the Filter field list. The following figure shows the Filter field list of the print server definition.

ORACLE® BI Publisher Enterprise

Welcome, administrator Dashboards Preferences Out Sign Help

Reports Schedules **Admin**

Admin > Printer > Update Server: hp8000-5n

Update Server: hp8000-5n Cancel Apply

General

Server Name: hp8000-5n * URI:
(Example: ipp://myhost:631/printers/myprinter)

Filter: Filter Command:
(Example: pdf2ps {infile} -{outfile})

Security

Username: Password:
 Authentication Type: Encryption Type:

For information on setting up the other fields in the Print Server definition page, see *Oracle Business Intelligence Publisher Report Designer's Guide*.

After completing all other required fields for the print server, you can schedule reports to print directly from the BI Publisher server to any printer that supports PostScript Level 2.

22.3 Limitations

This release of the PDF to PostScript converter has the following limitations:

- Only PDF version 1.4 or earlier is supported. This is also the currently supported version of the Oracle BI Publisher PDF output.
- CID fonts, which are used mainly to support languages with large sets of glyphs such as Chinese, Japanese, and Korean, are supported only when embedded in the PDF.
- Transparent and half-tone renderings are not supported.
- DeviceN color space is not supported.
- Shading patterns are not fully supported.
- Vertical writing is not supported.

Additional New Features in BI Publisher Release 10.1.3.3

This chapter describes features of BI Publisher that were introduced in Release 10.1.3.3 and that apply to Releases 10.1.3.3 through 10.1.3.4. Information about these features is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4. This chapter contains the following sections:

- [Section 23.1, "Support for CSV Output"](#)
- [Section 23.2, "Report Definition Features"](#)
- [Section 23.3, "Report Viewing Features"](#)
- [Section 23.4, "New Property for LDAP Security Model"](#)

23.1 Support for CSV Output

You can use Comma Separated Value (CSV) as an output type for any report with a simple <rowset>/<row> data structure. Support for this output type is based on the structure of the data. No template is required.

23.2 Report Definition Features

The following new features have been added to enhance options when defining reports:

- BI Publisher's data template data source provides native support for BLOB, CLOB, long, and raw data types.
- When defining parameters for a report, if you select Menu as the parameter type, then you have the option under the "Can select all" box to choose between "Null value passed" and "All values passed". Previously, selecting "all" passed a null value by default.
- The following properties are added to the Runtime Configuration page:
 - **RTF Output:** Default Font - enter the default font name and size (for example: Arial:12) to use when no font is defined. This is particularly useful for controlling the sizing of empty table cells in generated reports.
 - **HTML Output:** Use percentage width for table columns - set this property to True to render table columns according to a percentage value of the total width of the table rather than as a value in points.
 - **FO Processing:**

- * Disable external references - a "True" setting (default) disallows the importing of secondary files such as subtemplates or other XML documents during XSL processing and XML parsing. This increases the security of the system. Set this to "False" if the report or template calls external files.
- * FO Parsing Buffer Size - (default 1,000,000) sets the size of the buffer for the FO Processor. When the buffer is full, the elements from the buffer are rendered in the report. Reports with large tables or crosstabs might require a larger buffer to properly render those objects in the report. Increase the size of the buffer at the report level for these reports. Note that increasing this value affects the memory consumption of the system.

23.3 Report Viewing Features

The following new features are available from the View report page:

- Views created in the online Analyzer can be saved as templates.
- When in View mode, you see "Link to this Report" displayed on the page. This selection enables you to capture the URL of the currently viewed report. The Link to this Report selection has four options:
 - Current Page
 - No Header
 - No Parameters
 - Document Only

23.4 New Property for LDAP Security Model

For the LDAP security model, a new property, Group Retrieval Size, has been added. This property enables support of the LDAPv3 control extension for simple paging of search results. By default, pagination is not used. If this property is provided, then the value determines the number of results to return on a page (for example, 200).

The LDAP server must support control type 1.2.840.113556.1.4.319 to support this feature, such as OID 10.1.4. See your LDAP server documentation for information on support of this control type. For more information about LDAP pagination and the required control type, see the article: RFC 2696 - LDAP Control Extension for Simple Paged Results Manipulation (<http://www.faqs.org/rfcs/rfc2696.html>).

New Features in BI Publisher Release 10.1.3.3.1

This chapter describes features of BI Publisher that were introduced in Release 10.1.3.3.1 and that apply to Releases 10.1.3.3.1 through 10.1.3.4. Information about these features is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4. This chapter contains the following sections:

- [Section 24.1, "Time Out Value for Web Service Data Sources"](#)
- [Section 24.2, "Bidirectional Function for eText Data Fields"](#)
- [Section 24.3, "Default Repository"](#)

24.1 Time Out Value for Web Service Data Sources

You can now set a time out value when creating a data set based on a Web service. The Data Set Details definition page contains the "Time Out" field. The default is 60 seconds. Enter a numeric value to equal the number of seconds that BI Publisher waits for a response from the Web service before giving up.

To set the time out value for a Web service data set:

1. Navigate to the report, and select Edit.
2. Under the report Data Model region, select the Web service data set.
3. In the Details region, enter the desired time out period in seconds (default is 60 seconds).

24.2 Bidirectional Function for eText Data Fields

Starting with Release 10.1.3.3.1, you can use a BIDI function to preserve the correct order of characters for bidirectional languages in eText templates. If you have data fields that contain bidirectional data (for example, Arabic or Hebrew fields), then you can use the BIDI function to preserve the correct right to left ordering of the characters.

To use the BIDI function, enter the following in the template for each field that contains data to be treated as bidirectional:

BIDI (ELEMENT_NAME)

For example:

<POSITION>	<LENGTH>	<FORMAT>	<PAD>	<DATA>	<COMMENTS>
9	21	Alpha	L, ' '	BIDI (VENDOR_ADDRESS)	Vendor Address

In this example, the data for the element VENDOR_ADDRESS is treated as bidirectional.

24.3 Default Repository

BI Publisher now sets a default repository location.

The repository path is defined in the `xmlp-server-config.xml` file as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<xmlpConfig xmlns="http://xmlns.oracle.com/oxp/xmlp">
  <resource>
    <file path="${oracle.home}/xdo/repository"/>
  </resource>
</xmlpConfig>
```

If the "oracle.home" Java system property is set to a valid path, then `${oracle.home}` is replaced with that value.

When BI Publisher is deployed on OC4J, `${oracle.home}` is automatically set to a directory path where OC4J is installed.

For example, in an OC4J deployment in which the OC4J home is `C:\oc4j`, the BI Publisher repository is automatically set to:

`C:\oc4j\xdo\repository`

Note that the repository can also be set through BI Publisher Admin screens as shown in the following figure:

Figure 24–1 Setting the BI Publisher Report Repository Through the Admin Screen

The screenshot shows the Oracle BI Publisher Enterprise Admin interface. The top navigation bar includes 'Reports', 'Schedules', and 'Admin'. The 'Admin' tab is selected, and the breadcrumb trail shows 'Admin > Report Repository'. Below this, the 'System Maintenance' section has three tabs: 'Report Repository' (selected), 'Server Configuration', and 'Scheduler Configuration'. A tip message states: 'TIP Any changes will only take effect after the application is restarted.' There are 'Cancel' and 'Apply' buttons. The 'Repository Type' is set to 'File System' in a dropdown menu. The 'Path' field contains the value `${oracle.home}/xdo/repository`.

New Features in BI Publisher Release 10.1.3.3.2

This chapter describes two features of BI Publisher that were introduced in Release 10.1.3.3.2 and apply to Releases 10.1.3.3.2 through 10.1.3.4. Information about these features is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4. This chapter contains the following sections:

- [Section 25.1, "Support for Document Cache"](#)
- [Section 25.2, "Support for New Output Types"](#)

25.1 Support for Document Cache

In Release 10.1.3.3.2, BI Publisher introduces document caching. Document caching is enabled at the report level. With document cache enabled, when a user views the report online, the document (data plus layout) is placed in cache. When the same user uses the online Report Viewer to view the exact same report (same layout, same output type, same parameter selections) the document is retrieved from the cache. The document remains in the cache according to the cache specifications set in BI Publisher's System Maintenance page. Note that scheduled reports do not use the document cache.

To enable the document cache for a report, navigate to the report and select the **Edit** link. On the **Report Editor** page, view the **Report Properties**. Under **General Settings**, select **Enable Document Cache**.

For information on setting the cache specifications, see *Oracle Business Intelligence Publisher Report Designer's Guide*.

25.2 Support for New Output Types

Release 10.1.3.3.2 of BI Publisher introduces the following changes to the report output types available:

- **MHTML** - (Mime HyperText Markup Language) this format enables you to save a Web page and its resources as a single MHTML file (.mht), in which all images and linked files are saved as a single entity. Use this option to send or save HTML output and retain the embedded images and style sheet formatting.
- **Excel2000** - choose this output type if you have Microsoft Excel 2000 or 2002. Note that although Excel2000 is a new choice, it is actually the Excel option that was available in previous releases. This option generates HTML that can be opened in Excel. It does not support embedded images.

- Excel - choose this output type if you have Microsoft Excel 2003 or later installed. This option generates an MHTML format file that can be opened in Excel 2003 or later. This option supports embedded images, such as charts and logos.

New Features in BI Publisher Release 10.1.3.3.3

This chapter describes features of BI Publisher that were introduced in Release 10.1.3.3.3 and apply to Releases 10.1.3.3.3 through 10.1.3.4. Information about these features is also provided in the BI Publisher documentation set that has been updated for Release 10.1.3.4. This chapter contains the following sections:

- [Section 26.1, "BI Publisher Scheduler Setup Updates Including the Bundling of DataDirect Connect for JDBC Drivers"](#)
- [Section 26.2, ""Link to This Report" Can Now Be Set as a Report Option"](#)
- [Section 26.3, "Enhanced Support for Pagination Control in RTF Output"](#)
- [Section 26.4, "Support for Multiple Cascading Parameters"](#)
- [Section 26.5, "Support for Vertical Alignment of External Graphics"](#)
- [Section 26.6, "Support for Multiple Currency Symbol Displays in Report"](#)

26.1 BI Publisher Scheduler Setup Updates Including the Bundling of DataDirect Connect for JDBC Drivers

Release 10.1.3.3.3 of BI Publisher includes the Hyperion-branded DataDirect Connect for JDBC drivers that you can use to set up connection to the BI Publisher scheduler database. These drivers can be used as an alternative to the native JDBC drivers provided by the database vendor. Because the DataDirect drivers are now packaged with BI Publisher, you no longer must download and install the drivers manually.

DataDirect Connect for JDBC drivers are provided for the following databases:

- IBM DB2 v8.1, v9.1
- Microsoft SQL Server 2000, 2005
- Oracle 9i, Oracle 10g

Also in this release, the BI Publisher Scheduler Configuration page has been enhanced to facilitate the setup of the supported scheduler databases.

26.1.1 Recommendations for Using DataDirect Connect or Native Database Drivers

The following table displays the driver recommendations for the supported scheduler databases:

Database	Native JDBC Driver	DataDirect JDBC Driver
Oracle 9i, 10g	Recommended	Supported
IBM DB2 v8.1, v9.1	Supported	Recommended
Microsoft SQL Server 2000, 2005	Supported	Recommended
MySQL 4.1.10a-NT, 5.0	Supported	Not Supplied

26.1.2 Setting Up a Connection to Your Scheduler Database Using the DataDirect Connect for JDBC Drivers

Perform the following steps to set up your scheduler database using the DataDirect Connect for JDBC drivers:

1. Log in to BI Publisher with Administrator credentials and select the Admin tab.
2. Under **System Maintenance**, select **Scheduler Configuration**.
3. Enter the database connection information for the database using the appropriate connection string and database driver class as shown in the following sections.
4. Enter the **Username** and **Password** for the database. You must have permissions to connect to the database and create tables.
5. Click **Test Connection** to ensure that BI Publisher can connect to the database. If the connection fails, then ensure that you have entered the fields as shown and set up the database appropriately.
6. Click **Install Schema** to install the BI Publisher scheduler schema to the database.
7. Restart the BI Publisher application.
8. (Conditional) If the scheduler database is IBM DB2 or MySQL, set the size of the BLOB columns in the newly created scheduler tables in the database. See [Section 26.1.3, "Setting the BLOB Column Size."](#)

26.1.2.1 Entering Connection Information for IBM DB2

Enter the following if the scheduler database is IBM DB2 v8 or IBM DB2 v9:

Note: Ensure that the user that you enter to configure the scheduler has been set up with a 32 K page size tablespace. If not, create the table and assign it to the user. The user must also have "Connect to database" and "Create tables" privileges.

- Database Type: Select IBM DB2 v9 or IBM DB2 v8 from the list.
- Connection String: Enter the following connection string parameters:

```
jdbc:hyperion:db2://<hostname>:<port>;DatabaseName=<DATABASENAME>
```

Example:

```
jdbc:hyperion:db2:// 130.35.50.180:50000;DatabaseName=SAMPLE
```
- Database Driver Class: hyperion.jdbc.db2.DB2Driver

26.1.2.2 Entering Connection Information for Microsoft SQL Server

Enter the following if the scheduler database is Microsoft SQL Server 2000 or 2005.

Note: Ensure that the Microsoft SQL Server is set up with mixed mode authentication. Also ensure that the user that you enter to configure the scheduler has the "db_owner" role.

- Database Type: Select Microsoft SQL Server 2000 or Microsoft SQL Server 2005 from the list.
- Connection String: Enter the following connection string parameters:
`jdbc:hyperion:sqlserver://<hostname>:<port>;DatabaseName=<DatabaseName>`
Example:
`jdbc:hyperion:sqlserver://my.databaseserver.com:1433;DatabaseName=BIPublisher`
- Database Driver Class: `hyperion.jdbc.sqlserver.SQLServerDriver`

26.1.2.3 Entering Connection Information for Oracle

Enter the following if the scheduler database is Oracle 10g or Oracle 9i.

Note: Ensure that the database user that you enter has "connect" or "create session" and "create table" privileges and that the user has been assigned a quota (for example, 100M on USERS, 1M on TEMP).

- Database Type: Select Oracle 10g or Oracle 9i from the list.
- Connection String: Enter the following connection string parameters:
`jdbc:oracle:thin:@<hostname>:<port#>:<oracle SID>`
Example:
`jdbc:oracle:thin:@bipub.mycompany.com:1521:bipub`
- Database Driver Class: `oracle.jdbc.driver.OracleDriver`

26.1.2.4 Entering Connection Information for MySQL

Enter the following if the scheduler database is MySQL 4.1 or MySQL 5.0

Note: You must download the MySQL JDBC driver and register it with the application server where BI Publisher is running. The MySQL JDBC driver can be downloaded from:

<http://www.mysql.com/products/connector/j/>

See your application server documentation for information on registering the driver.

- Database Type: Select MySQL 4.1 or MySQL 5.0 from the list.
- Connection String: Enter the following connection string parameters:

```
jdbc:mysql://<hostname>:<port#>:<database name>
```

Example:

```
jdbc:mysql://localhost:3306/bipublisher
```

- Database Driver Class: com.mysql.jdbc.Driver

26.1.3 Setting the BLOB Column Size

When BI Publisher installs the scheduler schema to the database, it creates a table called XMLP_SCHED_OUTPUT. This table contains the BLOB columns that hold the BI Publisher scheduler output and data. It is recommended that the BLOB columns are set to be at least 2 GB, otherwise the scheduler may not be able to process large reports. If you are installing the scheduler schema to an IBM DB2 or MySQL database, then you must update the size of the BLOB columns listed below.

Following are the BLOB columns in the XMLP_SCHED_OUTPUT table. Reset the size of these columns to at least 2 GB.

- XML_DATA
- DOCUMENT_DATA

The following example shows the command to update the column size for an IBM DB2 database:

```
alter table xmlp_sched_output alter column STATUS_DETAIL set data type  
BLOB(1000000000);
```

See your database documentation for instructions on resetting the column size.

26.2 "Link to This Report" Can Now Be Set as a Report Option

When you view a report in BI Publisher's report viewer, a "Link to this Report" link displays enabling you to copy the URL of the current report. Previously the display of this link could not be disabled. In Release 10.1.3.3.3, this feature is now a report property, "Show Report Links" that can be set on the General Settings page of the Report definition. This enhancement enables you to choose whether you want the report link available to users.

By default, the "Show Report Links" property is enabled.

To change the setting for a report:

1. Log in to BI Publisher with Administrator or Report Developer credentials.
2. Navigate to the desired report and click the **Edit** link.
3. In the **Report Properties** pane of the report definition, under **General Settings**, select the **Show Report Links** box to enable the "Link to this Report" feature for the report. Deselect the box to disable the feature for the report.

26.3 Enhanced Support for Pagination Control in RTF Output

Previously BI Publisher recognized the following pagination control commands for PDF output only.

- Keep paragraph with the next paragraph
- Keep paragraph intact
- Table heading rows repeat

- Table row keep together

In Release 10.1.3.3.3, these are supported for RTF output as well.

26.3.1 Support for "Keep Paragraph with the Next Paragraph"

The RTF standard includes a command that enables you to "keep a paragraph with the next paragraph." This command prevents a page break between the paragraph for which it is enabled and the following (or next) paragraph.

To enable "keep paragraph with the next paragraph" in Microsoft Word:

1. Select the paragraphs you want to keep on a page.
2. From the **Format** menu, select **Paragraph**, and then click the **Line and Page Breaks** tab.
3. Select the **Keep with Next** box.

26.3.2 Keep Paragraph Intact

The RTF standard includes a command that enables you to keep all the lines of a paragraph joined ("keep paragraph intact") and prevent the paragraph from breaking across pages.

To enable "keep paragraph intact" in Microsoft Word:

1. From the **Format** menu, select **Paragraph**, and then select the **Line and Page Breaks** tab.
2. Deselect the **Keep lines together** check box.

26.3.3 Support for Table "Heading Rows Repeat"

The RTF standard includes a command that enables the repeat of the table heading rows when the table breaks across pages.

To enable "heading rows repeat" in Microsoft Word:

1. Select the heading row or rows.
2. From the **Table** menu, select **Heading Rows Repeat**.

26.3.4 Support for "Table Row Keep Together"

The RTF standard includes a command that enables you to prevent a table row from breaking across pages.

To enable "table row keep together" in Microsoft Word:

1. Select the table.
2. From the **Table** menu, select **Table Properties**, and then select the **Row** tab.
3. Deselect the **Allow row to break across pages** check box.

26.4 Support for Multiple Cascading Parameters

Cascading parameters enable you to filter available values for each parameter in a set of parameters that have a relationship. The choice that you make from the first parameter in a set of cascading parameters determines the available values for the next parameter. In this way, you can filter choices down to a manageable number for each cascading parameter. Previously only two levels of cascading parameters could be

supported. In Release 10.1.3.3.3, BI Publisher supports cascading parameters beyond two levels.

26.5 Support for Vertical Alignment of External Graphics

Support for the attribute "vertical-align" for the FO element "external-graphic" has been added in Release 10.1.3.3.3 of BI Publisher. This support enables you to format the alignment of an externally referenced image inline with text. This attribute can be applied when coding an XSL-FO template or an RTF template.

The valid values for the attribute are:

- bottom
- top
- middle
- text-top
- text-bottom

Follow the guidelines in *Oracle Business Intelligence Publisher Report Designer's Guide* for inserting an image using a URL reference:

1. In the template, insert a dummy image.
2. Select the image and open Microsoft Word's **Format Picture** dialog. Select the **Web** tab.
3. In the **Alternative text** region, enter the following syntax to reference the image URL:

```
url: {'http://image location'}
```

For example:

```
url: {'http://www.oracle.com/images/ora_logo.gif'}
```

4. To format the vertical alignment, declare the XSL attribute value before the URL. For example, to align the image to the bottom of the text (text-bottom), enter the following:

```
<xsl:attribute  
name="vertical-align">text-bottom</xsl:attribute>  
url: {'http://www.oracle.com/images/ora_logo.gif'}
```

26.6 Support for Multiple Currency Symbol Displays in Report

The format-currency function now accepts a third parameter, "displaySymbolOrNot" to enable the display of multiple currency codes in a single report. The International Standards Organization (ISO) currency code must be present in the report data to pass to the format-currency function.

The parameters for the format-currency function are as follows:

```
<?format-currency: Amount ; CurrencyCode ; displaySymbolOrNot?>
```

where

Amount takes the tag name of the XML element that holds the amount value in the data.

CurrencyCode takes the tag name of the XML element that holds the ISO three-letter currency code. An element that contains the currency code must be present in the data.

displaySymbolOrNot takes one of the following values: "true" or "false". When set to "true", the currency symbol is displayed in the report based on the value for CurrencyCode.

To avoid displaying the currency symbol, you can either enter "false" or simply do not specify the parameter.

26.6.1 Sample Usage

The following simple XML sample includes an element that contains the Amount (Trans_Amount) and an element that contains the ISO currency code (Cur_Code):

```
<ROW>
<Trans_Amount>123</Trans_Amount>
<Cur_Code>USD</Cur_Code>
</ROW>
<ROW>
<Trans_Amount>-456</Trans_Amount>
<Cur_Code>GBP</Cur_Code>
</ROW>
<ROW>
<Trans_Amount>748</Trans_Amount>
<Cur_Code>EUR</Cur_Code>
</ROW>
<ROW>
<Trans_Amount>-987</Trans_Amount>
<Cur_Code>JPY</Cur_Code>
</ROW>
```

To display each of these amounts with the appropriate currency symbol, enter the following in the template for the field in which you want the amounts to display:

```
<?format-currency:Trans_Amount;Cur_Code;'true'?>
```

The following figure shows sample output that can be achieved:

USD Amount:	\$123.00
GBP Amount:	-£456.00
EUR Amount:	€ 748.00
JPY Amount:	-¥987

For more information on using the format-currency function, see the Oracle BI Publisher Blog entry, "Multi-Currency Reports" available here:

http://blogs.oracle.com/xmlpublisher/entry/multi_currency_reports

For more information about the ISO currency codes, see the ISO Web site:

<http://www.iso.org/iso/home.htm>

BI Publisher Web Services, Releases 10.1.3.3.1, 10.1.3.3.2, and 10.1.3.3.3

This chapter explains the Oracle BI Publisher Web Services APIs and related information for Releases 10.1.3.3.1 through 10.1.3.3.3. Information about the complete set of Oracle BI Publisher Web Services APIs that you can use with Release 10.1.3.4 is provided in the chapter: "Using the BI Publisher Web Service APIs" in the *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

This chapter assumes a basic understanding of Web services and application programming interfaces. It contains the following sections:

- [Section 27.1, "Overview of the Oracle BI Publisher Web Services"](#)
- [Section 27.2, "Oracle BI Publisher Web Services Data Types"](#)
- [Section 27.3, "Oracle BI Publisher Web Services Operations"](#)
- [Section 27.4, "Debugging Web Service Applications"](#)
- [Section 27.5, "Oracle BI Publisher Web Services WSDL"](#)

27.1 Overview of the Oracle BI Publisher Web Services

This section provides an overview of the Oracle BI Publisher Web Services.

27.1.1 Functionality Overview

PublicReportService is the Web service that wraps the Oracle BI Publisher public APIs. Starting with Release 10.1.3.3.1, the following operations are supported. These operations can be grouped into three categories:

- Operations for validation of privileges
- Operations to get information about reports and the repository
- Operations to run and schedule reports

27.1.2 WSDL Definition for PublicReportService

After you have installed or deployed Oracle BI Publisher, there is a unique URL associated with PublicReportService:

```
http://<host>:<port>/xmlpserver/services/PublicReportService?wsdl
```

Enter this URL in the browser, substituting in the correct host and port number, to display the full Web Service Description Language definition for the publicly supported BI Publisher Web service.

On some J2EE servers, this URL might not correctly display the WSDL for this Web service. See the section [Section 27.5, "Oracle BI Publisher Web Services WSDL"](#) to see the full definition.

27.2 Oracle BI Publisher Web Services Data Types

This section covers the following topics:

- Base Data Types
- XML-to-Java Data Type Mappings
- Complex Types

27.2.1 Base Data Types

Oracle BI Publisher Web services use the following base data types:

- `xsd:boolean` = Boolean (example: true, false)
- `xsd:dateTime` = Date and Time (example: 2007-10-26T21:32:52)
- `xsd:int` = Integer (example: 23)
- `xsd:string` = String (example: /Home/Shared/HR Reports/Salary Report)
- `xsd:base64Binary` = 64 bit binary (example: A document in a format such as PDF or HTML)

27.2.2 XML-to-Java Data Type Mappings

BI Publisher Web Services use RPC-encoded formats. The mapping between Web service XML schema data types and Java data types depends on the SOAP development environment. The following list shows mappings for XML schema data types and the Oracle JDeveloper ones:

- `xsd:boolean` = `java.lang.Boolean`
- `xsd:dateTime` = `java.util.Date`
- `xsd:int` = `java.lang.Integer`
- `xsd:string` = `java.lang.String`
- `xsd:base64Binary` = `java.lang.Byte`

27.2.3 Complex Types

Oracle BI Publisher Web services define and use the following complex types:

ArrayOf_xsd_string

```
<complexType name="ArrayOf_xsd_string">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="xsd:string[]" />
    </restriction>
  </complexContent>
</complexType>
```

Description: an array of strings used in other complex types.

ParamNameValue

```
<complexType name="ParamNameValue">
  <sequence>
    <element name="multiValuesAllowed" type="xsd:boolean" />
    <element name="name" nillable="true" type="xsd:string" />
    <element name="values" nillable="true" type="impl:ArrayOf_xsd_string" />
  </sequence>
</complexType>
```

Description: a parameter name and array of parameter values.

ArrayOfParamNameValue

```
<complexType name="ArrayOfParamNameValue">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="impl:ParamNameValue[]" />
    </restriction>
  </complexContent>
</complexType>
```

Description: an array of ParamNameValues.

ReportDefinition

```
<complexType name="ReportDefinition">
  <sequence>
    <element name="defaultOutputFormat" nillable="true" type="xsd:string" />
    <element name="defaultTemplateId" nillable="true" type="xsd:string" />
    <element name="parameterNames" nillable="true"
      type="impl:ArrayOf_xsd_string" />
    <element name="reportParameterNameValues" nillable="true"
      type="impl:ArrayOfParamNameValue" />
    <element name="templateIds" nillable="true" type="impl:ArrayOf_xsd_string" />
  </sequence>
</complexType>
```

Description: object returned from getReportDefinition.

ItemData

```
<complexType name="ItemData">
  <sequence>
    <element name="absolutePath" nillable="true" type="xsd:string" />
    <element name="creationDate" nillable="true" type="xsd:dateTime" />
    <element name="displayName" nillable="true" type="xsd:string" />
    <element name="fileName" nillable="
true" type="xsd:string" />
    <element name="lastModified" nillable="true" type="xsd:dateTime" />
    <element name="lastModifier" nillable="true" type="xsd:string" />
    <element name="owner" nillable="true" type="xsd:string" />
    <element name="parentAbsolutePath" nillable="true" type="xsd:string" />
    <element name="type" nillable="true" type="xsd:string" />
  </sequence>
</complexType>
```

Description: description of an object contained in the report repository.

ArrayOfItemData

```
<complexType name="ArrayOfItemData">
  <complexContent>
```

```
<restriction base="soapenc:Array">
  <attribute ref="soapenc:arrayType"
wsdl:arrayType="impl:ItemData[]" />
</restriction>
</complexContent>
</complexType>
```

Description: an array of ItemData - objects contained in the report repository.

ReportRequest

```
<complexType name="ReportRequest">
  <sequence>
    <element name="attributeFormat" nillable="true" type="xsd:string"/>
    <element name="attributeLocale" nillable="true" type="xsd:string"/>
    <element name="attributeTemplate" nillable="true" type="xsd:string"/>
    <element name="parameterNameValues" nillable="true"
      type="impl:ArrayOfParamNameValue"/>
    <element name="reportAbsolutePath" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>
```

Description: the collection of settings needed to run a report. Note that allowable values for attributeFormat vary according to the type of template used (for example, PDF templates can only generate PDF output.)

ReportResponse

```
<complexType name="ReportResponse">
  <sequence>
    <element name="reportBytes" nillable="true" type="xsd:base64Binary" />
    <element name="reportContentType" nillable="true" type="xsd:string" />
    <element name="reportLocale" nillable="true" type="xsd:string" />
  </sequence>
</complexType>
```

Description: the document (in whatever file format attributeFormat is set to in ReportRequest), document type, and document locale that is returned from the runReport operation.

EmailDeliveryOption

```
<complexType name="EmailDeliveryOption">
  <sequence>
    <element name="emailBCC" nillable="true" type="xsd:string"/>
    <element name="emailBody" nillable="true" type="xsd:string"/>
    <element name="emailCC" nillable="true" type="xsd:string"/>
    <element name="emailFrom" nillable="false" type="xsd:string"/>
    <element name="emailReplyTo" nillable="true" type="xsd:string"/>
    <element name="emailSubject" nillable="false" type="xsd:string"/>
    <element name="emailTo" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>
```

Description: the options to set for e-mail delivery of a report.

FaxDeliveryOption

```
<complexType name="FaxDeliveryOption">
  <sequence>
    <element name="faxNumber" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>
```

```

    <element name="faxServer" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>

```

Description: the options to set for facsimile (fax) delivery of a report.

FTPDeliveryOption

```

<complexType name="FTPDeliveryOption">
  <sequence>
    <element name="ftpServerName" nillable="false" type="xsd:string"/>
    <element name="ftpUserName" nillable="false" type="xsd:string"/>
    <element name="ftpUserPassword" nillable="false" type="xsd:string"/>
    <element name="remoteFile" nillable="false" type="xsd:string"/>
    <element name="sftpOption" type="xsd:boolean"/>
  </sequence>
</complexType>

```

Description: the options to set for FTP delivery of a report.

LocalDeliveryOption

```

<complexType name="LocalDeliveryOption">
  <sequence>
    <element name="destination" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>

```

Description: the options to set for delivery of a report to the BI Publisher repository.

PrintDeliveryOption

```

<complexType name="PrintDeliveryOption">
  <sequence>
    <element name="printNumberOfCopy" nillable="true" type="xsd:string"/>
    <element name="printRange" nillable="true" type="xsd:string"/>
    <element name="printSide" nillable="true" type="xsd:string"/>
    <element name="printTray" nillable="true" type="xsd:string"/>
    <element name="printerName" nillable="false" type="xsd:string"/>
  </sequence>
</complexType>

```

Description: the options to set for printer delivery of a report.

WebDAVDeliveryOption

```

<complexType name="WebDAVDeliveryOption">
  <sequence>
    <element name="deliveryAuthType" nillable="true" type="xsd:string"/>
    <element name="deliveryAuthTypeBasic" nillable="true" type="xsd:string"/>
    <element name="deliveryAuthTypeDigest" nillable="true" type="xsd:string"/>
    <element name="password" nillable="true" type="xsd:string"/>
    <element name="remoteFilePath" nillable="false" type="xsd:string"/>
    <element name="server" nillable="false" type="xsd:string"/>
    <element name="userName" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>

```

Description: the options to set for Web-based Distributed Authoring and Versioning (WebDAV) delivery of a report.

DeliveryRequest

```
<complexType name="DeliveryRequest">
  <sequence>
    <element name="emailOption" nillable="true" type="impl:EMailDeliveryOption"/>
    <element name="faxOption" nillable="true" type="impl:FaxDeliveryOption"/>
    <element name="ftpOption" nillable="true" type="impl:FTPDeliveryOption"/>
    <element name="localOption" nillable="true" type="impl:LocalDeliveryOption"/>
    <element name="printOption" nillable="true" type="impl:PrintDeliveryOption"/>
    <element name="webDAVOption" nillable="true"
      type="impl:WebDAVDeliveryOption"/>
  </sequence>
</complexType>
```

Description: an object that provides the options to deliver a report to multiple destinations.

ScheduleRequest

```
<complexType name="ScheduleRequest">
  <sequence>
    <element name="deliveryRequest" nillable="false"
      type="impl:DeliveryRequest"/>
    <element name="endDate" nillable="true" type="xsd:dateTime"/>
    <element name="jobCalendar" nillable="true" type="xsd:string"/>
    <element name="jobLocale" nillable="true" type="xsd:string"/>
    <element name="jobTZ" nillable="true" type="xsd:string"/>
    <element name="notificationTo" nillable="true" type="xsd:string"/>
    <element name="notifyWhenFailed" type="xsd:boolean" />
    <element name="notifyWhenSuccess" type="xsd:boolean" />
    <element name="notifyWhenWarning" type="xsd:boolean" />
    <element name="repeatCount" type="xsd:int"/>
    <element name="repeatInterval" type="xsd:int"/>
    <element name="reportRequest" nillable="false" type="impl:ReportRequest"/>
    <element name="saveDataOption" type="xsd:boolean" />
    <element name="saveOutputOption" type="xsd:boolean" />
    <element name="scheduleBurstringOption" type="xsd:boolean" />
    <element name="schedulePublicOption" type="xsd:boolean" />
    <element name="startDate" nillable="true" type="xsd:dateTime"/>
    <element name="useUTF8Option" type="xsd:boolean" />
    <element name="userJobName" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
```

Description: the options to schedule a report.

AccessDeniedException

```
<complexType name="AccessDeniedException">
  <sequence />
</complexType>
```

Description: message returned when the user credentials do not have adequate privileges to complete an operation.

OperationFailedException

```
<complexType name="OperationFailedException">
  <sequence />
</complexType>
```

Description: the message returned when an operation fails.

InvalidParametersException

```
<complexType name="InvalidParametersException">
  <sequence />
</complexType>
```

Description: message returned when invalid parameters are supplied for a Web service operation.

27.3 Oracle BI Publisher Web Services Operations

The operations available in the Oracle BI Publisher Web services can be described by the following categories:

- Operations for validation of privileges
- Operations to get information about reports and the repository
- Operations to run and schedule reports

This section describes the input message, output message, possible exception messages and also provides sample code for each operation.

27.3.1 About Using the Code Samples

Each operation listed in the following section includes sample code. The sample code is based on Apache Axis technology. Before working with the examples, ensure that you have included the prerequisite files in your classpath and generated the required BI Publisher stub classes to import into the project.

Include the Required jar Files

The following list describes the jar files to include:

- Include the following jar files in your classpath (these files can be obtained from the Web):
 - activation.jar
 - mail-1.4.jar
- Include `xmlpservice.jar` file (installed with Oracle BI Publisher) in your classpath. This specification ensures that you have all the required request and response objects. This way you need not generate response and request stub objects.

Generate the Required BI Publisher Stub Classes

To generate the required BI Publisher stub classes to import to the project for this sample, perform the following:

1. Download the BI Publisher Web services WSDL.
2. Run the following in a command file:

```
export CLASSPATH="D:/bin/axis-1_4/lib/axis.jar;D:/bin/axis-1_4/lib/jaxrpc.jar;D:/bin/axis-1_4/lib/saaj.jar;D:/bin/axis-1_4/lib/commons-logging.jar;D:/bin/axis-1_4/lib/commons-discovery.jar;D:/bin/axis-1_4/lib/wsdl4j.jar"
echo $CLASSPATH
java org.apache.axis.wsdl.WSDL2Java PublicReportService.wsdl
```

This command generates the following BI Publisher stub classes for you to import into your project:

```
import com.oracle.xmlns.oxp.service.PublicReportService.ItemData;
import com.oracle.xmlns.oxp.service.PublicReportService.ReportRequest;
import com.oracle.xmlns.oxp.service.PublicReportService.ReportResponse;
import com.oracle.xmlns.oxp.service.PublicReportService.ParamNameValue;
import com.oracle.xmlns.oxp.service.PublicReportService.ReportDefinition;
import com.oracle.xmlns.oxp.service.PublicReportService.ScheduleRequest;
import com.oracle.xmlns.oxp.service.PublicReportService.DeliveryRequest;
import com.oracle.xmlns.oxp.service.PublicReportService.EMailDeliveryOption;
```

Import the Required Apache Axis and Java Stub Classes

In addition to importing the BI Publisher Web service stub classes that you generated, you must also import the following Apache Axis and Java classes:

```
import java.io.FileOutputStream;
import java.io.OutputStream;
import java.util.Calendar;
import org.apache.axis.client.Call;
import org.apache.axis.client.Service;
import org.apache.axis.encoding.XMLType;
import org.apache.axis.encoding.ser.BeanDeserializerFactory;
import org.apache.axis.encoding.ser.BeanSerializerFactory;
import javax.xml.namespace.QName;
import javax.xml.rpc.ParameterMode;
import java.net.URL;
```

27.3.2 Operations for Validation of Privileges

validateLogin

Use validateLogin to validate that a UserID and Password have the privilege to access the Oracle BI Publisher report server.

input message = **validateLoginRequest**

```
<wsdl:message name="validateLoginRequest">
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>
```

output message = **validateLoginResponse**

```
<wsdl:message name="validateLoginResponse">
  <wsdl:part name="validateLoginReturn" type="xsd:boolean"/>
</wsdl:message>
```

Example 27–1 Sample Code for validateLogin

```
public static void validateLogin() throws Exception {

    final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";
    final String reportPath = "/HR Manager/Employee Salary Report/Employee Salary
Report.xdo";

    service = new Service();
```

```

call = (Call) service.createCall();
call.setTargetEndpointAddress(new URL(bipEndpoint));

System.out.println("TESTING login Service BEGIN");

call.setOperationName(new QName(bipNamespace, "validateLogin"));
call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
call.setReturnType(XMLType.XSD_BOOLEAN);

// issue the request
Boolean valid = (Boolean) call.invoke(
    new Object[] { "administrator", "Administrator" });

if (valid) { System.out.println("user valid"); }
else { System.out.println("user invalid"); }
System.out.println("Success for validateLogin");
}

```

hasReportAccess

Use `hasReportAccess` to validate that a `UserID` and `Password` have the privilege to access a specific report.

input message = `hasReportAccessRequest`

```

<wsdl:message name="hasReportAccessRequest">
  <wsdl:part name="reportAbsolutePath" type="xsd:string"/>
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>

```

output message = `hasReportAccessResponse`

```

<wsdl:message name="hasReportAccessResponse">
  <wsdl:part name="hasReportAccessReturn" type="xsd:boolean"/>
</wsdl:message>

```

Possible exceptions:

`InvalidParametersException`

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–2 Sample Code for `hasReportAccess`

```

public static void hasReportAccess(String reportPath) throws Exception {

    final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    //final String bipEndpoint =
"http://12.123.8232.107/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";

    service = new Service();
    call = (Call) service.createCall();
    call.setTargetEndpointAddress(new URL(bipEndpoint));

    System.out.println("TESTING hasReportAccess BEGIN");

    call.setOperationName(new QName(bipNamespace, "hasReportAccess"));
    call.addParameter("reportAbsolutePath", XMLType.XSD_STRING, ParameterMode.IN);
    call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
    call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
    call.setReturnType(XMLType.XSD_BOOLEAN);
}

```

```
// issue the request
Boolean valid = (Boolean) call.invoke(
    new Object[] {reportPath, "administrator", "Administrator"});

if (valid==true) { System.out.println("User has access."); }
else { System.out.println("User does not have access"); }
System.out.println("Success for AccessLogin");
}
```

27.3.3 Operations to Get Information About Reports and the Repository

getFolderContents

Use `getFolderContents` to get all of the items in a folder. This command returns all the reports and folders that are contained in the specified folder. You can then use these items to determine what reports you might want to execute or what folders you might want to further search to identify a report.

input message = **getFolderContentsRequest**

```
<wsdl:message name="getFolderContentsRequest">
  <wsdl:part name="folderAbsolutePath" type="xsd:string"/>
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>
```

output message = **getFolderContentsResponse**

```
<wsdl:message name="getFolderContentsResponse">
  <wsdl:part name="getFolderContentsReturn" type="impl:ArrayOfItemData"/>
</wsdl:message>
```

Possible exceptions:

`InvalidParametersException`
`AccessDeniedException`
`OperationFailedException`

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–3 Sample Code for `getFolderContents`

```
public static void getFolderContent(String reportPath) throws Exception {

    final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";

    service = new Service();
    call = (Call) service.createCall();
    call.setTargetEndpointAddress(new URL(bipEndpoint));

    System.out.println("TESTING getFolderContent Service BEGIN");
    // register the ReportDefinition class
    QName reportDef = new QName(bipNamespace, "ReportDefinition");
    call.registerTypeMapping(ReportDefinition.class, reportDef,
        BeanSerializerFactory.class, BeanDeserializerFactory.class);

    // register the ParamNameValue class
    QName nmvals = new QName(bipNamespace, "ParamNameValue");
    call.registerTypeMapping(ParamNameValue.class, nmvals, BeanSerializerFactory.class,
        BeanDeserializerFactory.class);
}
```



```

call.setOperationName(new QName(bipNamespace, "getFolderContents"));
call.addParameter("reportAbsolutePath", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
call.setReturnClass(ItemData [].class);

// issue the request
ItemData items[] = (ItemData []) call.invoke(
    new Object[] { reportPath, "administrator", "Administrator"});

for (int i=0; i<items.length; i++) {
    System.out.println(items[i].getDisplayName());
}
System.out.println("Success for GetFolderContent");
}

```

getReportDefinition

Use `getReportDefinition` to get information about a report such as default template, output type and a listing of template IDs. When you have a list of template IDs, you might want to generate a report with a template other than the default.

input message = **getReportDefinitionRequest**

```

<wsdl:message name="getReportDefinitionRequest">
  <wsdl:part name="reportAbsolutePath" type="xsd:string"/>
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>

```

output message = **getReportDefinitionResponse**

```

<wsdl:message name="getReportDefinitionResponse">
  <wsdl:part name="getReportDefinitionReturn" type="impl:ReportDefinition"/>
</wsdl:message>

```

Possible exceptions:

```

InvalidParametersException
AccessDeniedException
OperationFailedException

```

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–4 Sample Code for `getReportDefinition`

```

public static void getReportDef(String xdoFile) throws Exception {

    final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";

    service = new Service();
    call = (Call) service.createCall();
    call.setTargetEndpointAddress(new URL(bipEndpoint));

    System.out.println("BEGIN TESTING getReportDefinition");

    // register the ReportDefinition class
    QName reportDef = new QName(bipNamespace, "ReportDefinition");
    call.registerTypeMapping(ReportDefinition.class, reportDef,
        BeanSerializerFactory.class, BeanDeserializerFactory.class);
}

```

```

// register the ParamNameValue class
QName nmvals = new QName(bipNamespace, "ParamNameValue");
call.registerTypeMapping(ParamNameValue.class, nmvals, BeanSerializerFactory.class,
BeanDeserializerFactory.class);

call.setOperationName(new QName(bipNamespace, "getReportDefinition"));
call.addParameter("reportAbsolutePath", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
call.setReturnClass(ReportDefinition.class);

// issue the request
ReportDefinition reportDefn = (ReportDefinition) call.invoke(
    new Object[] { xdofile, "administrator", "Administrator"});

System.out.println("Report Definition Returns with \n Default Output Format = " +
reportDefn.getDefaultOutputFormat());
ParamNameValue params [] = reportDefn.getReportParameterNameValues();

if (params != null) {
    for (int i = 0; i < params.length; i++) {
        System.out.print("Parameter " + params[i].getName() + " :");
        if (params[i].getValues() != null) {
            for (int j = 0; j < params[i].getValues().length; j++)
                System.out.print(" " + params[i].getValues()[j]);
        } else
            System.out.print(" null");
        System.out.println(" - multiple values? " +
params[i].isMultiValuesAllowed());
    }
}
System.out.println("END TESTING getReportDefinition");
}

```

getReportParameters

Use `getReportParameters` to get an array of report parameters and their default values. When you have the list of parameters, you can set parameter values before running or scheduling a report.

input message = **getReportParametersRequest**

```

<wsdl:message name="getReportParametersRequest">
  <wsdl:part name="reportRequest" type="impl:ReportRequest"/>
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>

```

output message = **getReportParametersResponse**

```

<wsdl:message name="getReportParametersResponse">
  <wsdl:part name="getReportParametersReturn"
    type="impl:ArrayOfParamNameValue"/>
</wsdl:message>

```

Possible exceptions:

```

InvalidParametersException
AccessDeniedException
OperationFailedException

```

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–5 Sample Code for `getReportParameters`

```

public static void getParameters(String xdofile) throws Exception {

```

```

        //final String bipEndpoint =
"http://10.138.232.107/xmlpserver/services/PublicReportService?wsdl";
        final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
        final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";

        service = new Service();
        call = (Call) service.createCall();
        call.setTargetEndpointAddress(new URL(bipEndpoint));
        System.out.println("BEGIN TESTING: getParameters Service");

        // register the ReportRequest class
        QName reportReq = new QName(bipNamespace, "ReportRequest");
        call.registerTypeMapping(ReportRequest.class, reportReq,
            BeanSerializerFactory.class, BeanDeserializerFactory.class);

        // register the ParamNameValue class
        QName nmvals = new QName(bipNamespace, "ParamNameValue");
        call.registerTypeMapping(ParamNameValue.class, nmvals, BeanSerializerFactory.class,
            BeanDeserializerFactory.class);

        // Define parameters
        ParamNameValue[] paramNmVals = new ParamNameValue[2];
        paramNmVals[0] = new ParamNameValue(false, false, "dept", null) ;
        paramNmVals[1] = new ParamNameValue(false, false, "emp", new String[] {""}) ;
        ReportRequest req = new ReportRequest("pdf", "en-US", "Simple", paramNmVals,
xdofile);

        call.setOperationName(new QName(bipNamespace, "getReportParameters"));
        call.addParameter("ReportRequest", reportReq, ParameterMode.IN);
        call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
        call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
        call.setReturnClass(ParamNameValue [].class);

        // issue the request
        ParamNameValue params [] = (ParamNameValue []) call.invoke(
            new Object[] { req, "administrator", "Administrator"});

        if (params != null) {
            for (int i = 0; i < params.length; i++) {
                System.out.print("Parameter " + params[i].getName() + " :");
                if (params[i].getValues() != null) {
                    for (int j = 0; j < params[i].getValues().length; j++)
                        System.out.print(" " + params[i].getValues()[j]);
                } else
                    System.out.print(" null");
                System.out.println(" - multiple values? " +
params[i].isMultiValuesAllowed());
            }
        }
        System.out.println("END TESTING: getParameters");
    }
}

```

27.3.4 Operations to Run and Schedule Reports

runReport

Use runReport to execute a report and return a generated document. Note that the document is returned in the specified file format.

input message = **runReportRequest**

```
<wsdl:message name="runReportRequest">
```

```
<wsdl:part name="reportRequest" type="impl:ReportRequest"/>
<wsdl:part name="userID" type="xsd:string"/>
<wsdl:part name="password" type="xsd:string"/>
</wsdl:message>
```

output message = **runReportResponse**

```
<wsdl:message name="runReportResponse">
  <wsdl:part name="runReportReturn" type="impl:ReportResponse"/>
</wsdl:message>
```

Possible exceptions:

InvalidParametersException
AccessDeniedException
OperationFailedException

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–6 Sample Code for runReport

```
public static void runReport() throws Exception {

    final String bipEndpoint =
        "http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
        "http://xmlns.oracle.com/oxp/service/PublicReportService";
    final String xdoFile = "/HR Manager/Employee Salary Report/Employee Salary
    Report.xdo";

    // set up the call object
    service = new Service();
    call = (Call) service.createCall();
    call.setTargetEndpointAddress(new URL(bipEndpoint));

    // TEST Run Report
    System.out.println("TESTING runReport Service BEGIN");

    // register the ReportRequest class
    QName reportReq = new QName(bipNamespace, "ReportRequest");
    call.registerTypeMapping(ReportRequest.class, reportReq,
        BeanSerializerFactory.class, BeanDeserializerFactory.class);

    // register the ParamNameValue class
    QName nmvals = new QName(bipNamespace, "ParamNameValue");
    call.registerTypeMapping(ParamNameValue.class, nmvals, BeanSerializerFactory.class,
        BeanDeserializerFactory.class);

    // register the BIPReportResponse class
    QName reportRespqn = new QName(bipNamespace, "ReportResponse");
    call.registerTypeMapping(ReportResponse.class, reportRespqn,
        BeanSerializerFactory.class, BeanDeserializerFactory.class);

    // Default return type based on what we expect
    call.setOperationName( new QName(bipNamespace, "runReport" ));
    call.addParameter( "ReportRequest", reportReq, ParameterMode.IN );
    call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
    call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
    call.setReturnClass(ReportResponse.class);

    ParamNameValue[] paramNmVals = new ParamNameValue[2];
    paramNmVals[0] = new ParamNameValue(false, "dept", null );
    paramNmVals[1] = new ParamNameValue(false, "emp", new String[] { "*" } );
}
```

```

        ReportRequest req = new ReportRequest("pdf", "en-US", "Simple", paramNmVals,
        xdoFile);

        // issue the request
        ReportResponse reportResp = (ReportResponse) call.invoke(
            new Object[] {req, "Administrator", "Administrator"} );
        System.out.println("Return ReportResponse: ContentType = " +
        reportResp.getReportContentType());

        // Save the report
        byte[] binaryBytes = reportResp.getReportBytes();
        OutputStream out = new FileOutputStream("D:\\temp\\out.pdf");
        out.write(binaryBytes);
        out.close();
        System.out.println("Success for Run Report");
    }

```

scheduleReport

Use `scheduleReport` to schedule a report for execution and delivery to either printer, fax, email, WebDAV, ftp or simply save in the report repository. Jobs can be scheduled to run immediately, once, or on a recurring pattern and can have an end date to stop the recurrence. This operation returns JobID upon successfully scheduling the report job.

inputmessage = **scheduleReportRequest**

```

<wsdl:message name="scheduleReportRequest">
  <wsdl:part name="scheduleRequest" type="impl:ScheduleRequest"/>
  <wsdl:part name="userID" type="xsd:string"/>
  <wsdl:part name="password" type="xsd:string"/>
</wsdl:message>

```

outputmessage = **scheduleReportResponse**

```

<wsdl:message name="scheduleReportResponse">
  <wsdl:part name="scheduleReportReturn" type="xsd:string"/>
</wsdl:message>

```

Possible exceptions:

```

InvalidParametersException
AccessDeniedException
OperationFailedException

```

See [Section 27.3.5, "Messages for Errors"](#) for descriptions of the possible exceptions.

Example 27–7 Sample Code for `scheduleReport`

```

public static void scheduleReport() throws Exception {

    final String bipEndpoint =
"http://mycompany.com:9999/xmlpserver/services/PublicReportService?wsdl";
    final String bipNamespace =
"http://xmlns.oracle.com/oxp/service/PublicReportService";
    final String xdoFile = "/HR Manager/Employee Salary Report/Employee Salary
Report.xdo";

    // set up the call object
    service = new Service();
    call = (Call) service.createCall();
    call.setTargetEndpointAddress(new URL(bipEndpoint));
}

```

```

// TEST Run Report
System.out.println("TESTING scheduleReport Service BEGIN");

// register the ReportRequest class
QName reportReq = new QName(bipNamespace, "ReportRequest");
call.registerTypeMapping(ReportRequest.class, reportReq,
    BeanSerializerFactory.class, BeanDeserializerFactory.class);

// register the ParamNameValue class
QName nmvals = new QName(bipNamespace, "ParamNameValue");
call.registerTypeMapping(ParamNameValue.class, nmvals, BeanSerializerFactory.class,
    BeanDeserializerFactory.class);

// register the BIPReportResponse class (part of Schedule Request)
QName reportRespqn = new QName(bipNamespace, "ReportResponse");
call.registerTypeMapping(ReportResponse.class, reportRespqn,
    BeanSerializerFactory.class, BeanDeserializerFactory.class);

// register the Schedule Request class
QName scheduleReq = new QName(bipNamespace, "ScheduleRequest");
call.registerTypeMapping(ReportRequest.class, reportReq,
    BeanSerializerFactory.class, BeanDeserializerFactory.class);

// Default return type based on what we expect
call.setOperationName( new QName(bipNamespace, "scheduleReport" ));
call.addParameter( "ScheduleRequest", scheduleReq, ParameterMode.IN );
call.addParameter("userID", XMLType.XSD_STRING, ParameterMode.IN);
call.addParameter("password", XMLType.XSD_STRING, ParameterMode.IN);
call.setReturnType(XMLType.XSD_STRING);

// Define Report Request Object
ParamNameValue[] paramNmVals = new ParamNameValue[2];
paramNmVals[0] = new ParamNameValue(false, "dept", null );
paramNmVals[1] = new ParamNameValue(false, "emp", new String[] {""});
ReportRequest report = new ReportRequest("pdf", "en-US", "Simple", paramNmVals,
xdofile);

// Define Email Delivery Options
EmailDeliveryOption email =
    new EmailDeliveryOption("manager@mycompany.com", "Body", // bcc, boddy
        "manager@mycompany.com", // cc
        "bipublisher@oracle.com", "bipublisher@oracle.com", // from, reply to
        "Test Email Subject", "manager@mycompany.com" // subject, to
    );

// Define only email delivery
DeliveryRequest delivery = new DeliveryRequest(email, null, null, null, null, null);

ScheduleRequest req = new ScheduleRequest(
    delivery,                // delivery options
    null,                    // enddate (Calendar.getInstance())
    null,                    // job calendar
    "en-US", null,           // job locale, timezone
    null, false, false, false, // notifications
    1, 1,                    // repeat count, repeat interval
    report,                  // Report and Parameters
    "true", "true",          // Save Data, Save Output
    "false", "true",         // Bursting, Public
    null,                    // start date
    "true",                  // Use UTF-8
    "my Job4"                // job name
);

// issue the request
String result = (String) call.invoke(
    new Object[] {req, "Administrator", "Administrator"} );

```

```

        System.out.println("Success for Schedule Report");
    }

```

27.3.5 Messages for Errors

One of the following messages may be returned to any of the operations if there is an error that occurs in the execution of the operation.

OperationFailedException

```

<wsdl:message name="OperationFailedException">
  <wsdl:part name="fault" type="impl:OperationFailedException"/>
</wsdl:message>

```

AccessDeniedException

```

<wsdl:message name="AccessDeniedException">
  <wsdl:part name="fault" type="impl:AccessDeniedException"/>
</wsdl:message>

```

InvalidParametersException

```

<wsdl:message name="InvalidParametersException">
  <wsdl:part name="fault" type="impl:InvalidParametersException"/>
</wsdl:message>

```

27.4 Debugging Web Service Applications

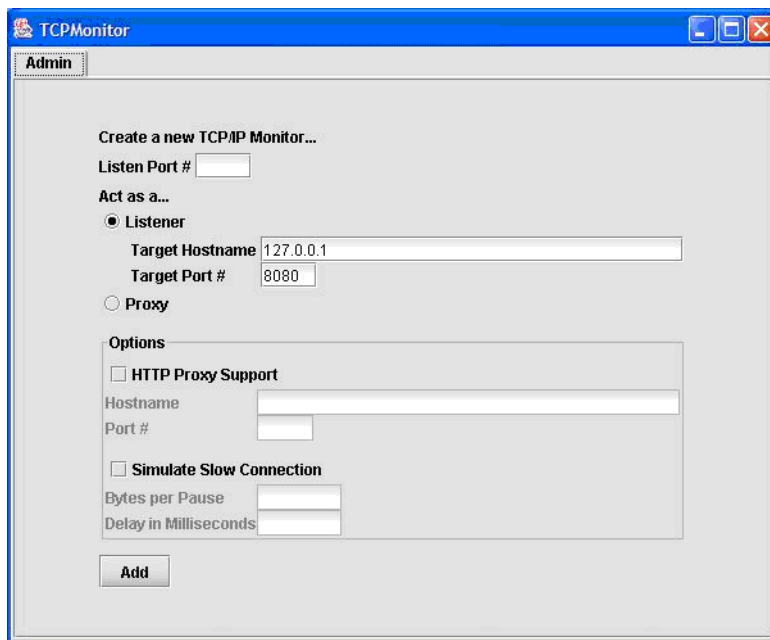
As a Web service developer you may need to see the SOAP request messages being used to invoke Web services along with the SOAP response to those request messages. Oracle BI Publisher bundles the Apache Axis SOAP Monitor utility to monitor the SOAP message flow without requiring any special configuration or restarting of the server.

The client sample code sends a SOAP request to listening port 8888, and then takes advantage of Axis TCP monitor to route SOAP request to port 8080, where the PublicReportService is running. This is a useful debugging tool for the Web service developer.

To start Axis TCP Monitor (tcpmon), from the command line enter the following:

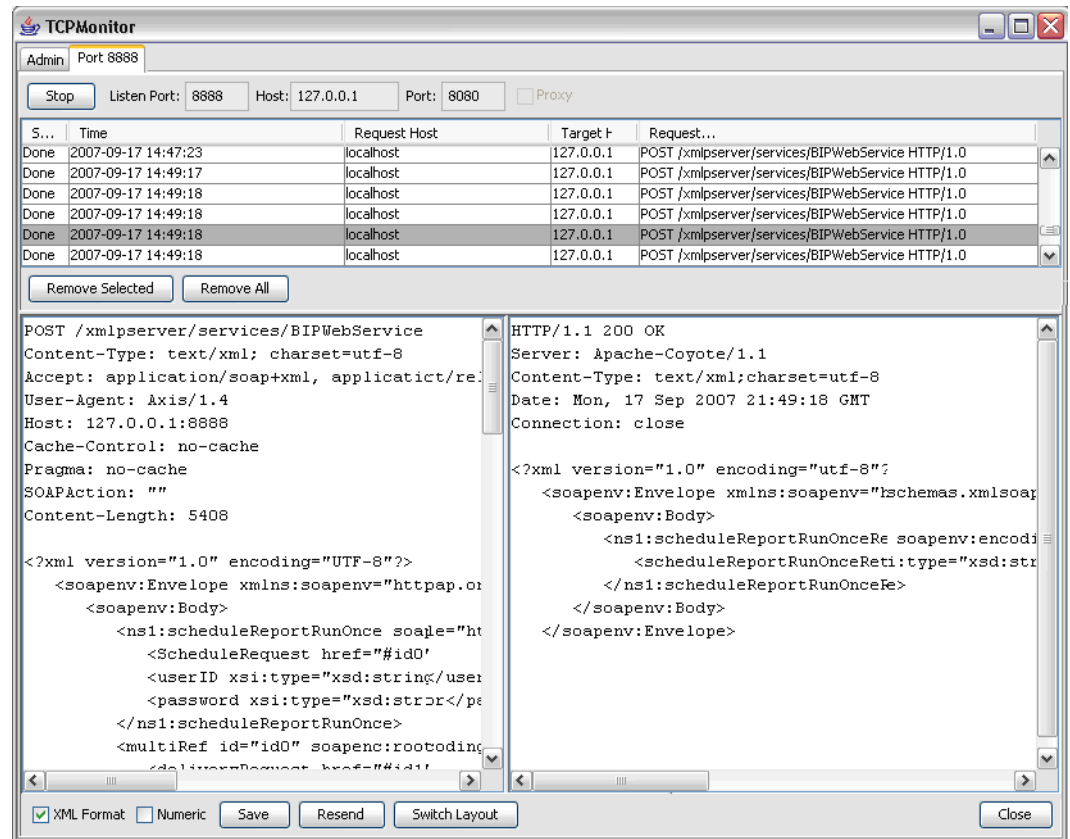
```
% java org.apache.axis.utils.tcpmon
```

This launches the TCP monitor window. A sample of the window without any of the optional arguments is shown in [Figure 27-1](#):

Figure 27–1 TCPMonitor Window

To use the program, select a local port that tcpmon monitors for incoming connections, a target host where it forwards such connections, and the port number on the target computer which should be "tunneled" to. Then click Add. Another tab is then displayed for the new tunneled connection. From there, you see the SOAP Request and Response message, which facilitates debugging.

Figure 27–2 shows the connection tab of the TCPMonitor window:

Figure 27–2 TCP Monitor Window Connection Tab

To run tcpmonitor, enter the following:

```
set CLASSPATH=.
set CLASSPATH=%CLASSPATH%;%WEB_HOME%\PORTAL\WEB-INF\lib\axis.jar
set TCPMON_PORT=8081
if not %1==. TCPMON_PORT=%1
java org.apache.axis.utils.tcpmon %TCPMON_PORT% %ADMINSERVER_HOSTNAME%
%ADMINSERVER_PORT%
```

27.5 Oracle BI Publisher Web Services WSDL

Following is the full BI Publisher PublicReportService Web Services WSDL. It is provided for those instances where BI Publisher has been deployed to a J2EE server that is not able to display this WSDL.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
targetNamespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://xmlns.oracle.com/oxp/service/PublicReportService"
xmlns:intf="http://xmlns.oracle.com/oxp/service/PublicReportService"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<!--WSDL created by Apache Axis version: 1.3
Built on Oct 05, 2005 (05:23:37 EDT)-->
<wsdl:types>
```

```

<schema
targetNamespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
xmlns="http://www.w3.org/2001/XMLSchema">
  <import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
  <complexType name="ArrayOf_xsd_string">
    <complexContent>
      <restriction base="soapenc:Array">
        <attribute ref="soapenc:arrayType" wsdl:arrayType="xsd:string[]" />
      </restriction>
    </complexContent>
  </complexType>
  <complexType name="ParamNameValue">
    <sequence>
      <element name="multiValuesAllowed" type="xsd:boolean" />
      <element name="name" nillable="true" type="xsd:string" />
      <element name="values" nillable="true" type="impl:ArrayOf_xsd_string" />
    </sequence>
  </complexType>
  <complexType name="ArrayOfParamNameValue">
    <complexContent>
      <restriction base="soapenc:Array">
        <attribute ref="soapenc:arrayType" wsdl:arrayType="impl:ParamNameValue[]" />
      </restriction>
    </complexContent>
  </complexType>
  <complexType name="ReportRequest">
    <sequence>
      <element name="attributeFormat" nillable="true" type="xsd:string" />
      <element name="attributeLocale" nillable="true" type="xsd:string" />
      <element name="attributeTemplate" nillable="true" type="xsd:string" />
      <element name="parameterNameValues" nillable="true"
type="impl:ArrayOfParamNameValue" />
      <element name="reportAbsolutePath" nillable="true" type="xsd:string" />
    </sequence>
  </complexType>
  <complexType name="AccessDeniedException">
    <sequence />
  </complexType>
  <complexType name="OperationFailedException">
    <sequence />
  </complexType>
  <complexType name="InvalidParametersException">
    <sequence />
  </complexType>
  <complexType name="ReportResponse">
    <sequence>
      <element name="reportBytes" nillable="true" type="xsd:base64Binary" />
      <element name="reportContentType" nillable="true" type="xsd:string" />
      <element name="reportLocale" nillable="true" type="xsd:string" />
    </sequence>
  </complexType>
  <complexType name="ReportDefinition">
    <sequence>
      <element name="defaultOutputFormat" nillable="true" type="xsd:string" />
      <element name="defaultTemplateId" nillable="true" type="xsd:string" />
      <element name="parameterNames" nillable="true" type="impl:ArrayOf_xsd_
string" />
      <element name="reportParameterNameValues" nillable="true"
type="impl:ArrayOfParamNameValue" />
      <element name="templateIds" nillable="true" type="impl:ArrayOf_xsd_string" />
    </sequence>
  </complexType>

```

```

</sequence>
</complexType>
<complexType name="ItemData">
  <sequence>
    <element name="absolutePath" nillable="true" type="xsd:string"/>
    <element name="creationDate" nillable="true" type="xsd:dateTime"/>
    <element name="displayName" nillable="true" type="xsd:string"/>
    <element name="fileName" nillable="true" type="xsd:string"/>
    <element name="lastModified" nillable="true" type="xsd:dateTime"/>
    <element name="lastModifier" nillable="true" type="xsd:string"/>
    <element name="owner" nillable="true" type="xsd:string"/>
    <element name="parentAbsolutePath" nillable="true" type="xsd:string"/>
    <element name="type" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
<complexType name="ArrayOfItemData">
  <complexContent>
    <restriction base="soapenc:Array">
      <attribute ref="soapenc:arrayType" wsdl:arrayType="impl:ItemData[]" />
    </restriction>
  </complexContent>
</complexType>
<complexType name="EMailDeliveryOption">
  <sequence>
    <element name="emailBCC" nillable="true" type="xsd:string"/>
    <element name="emailBody" nillable="true" type="xsd:string"/>
    <element name="emailCC" nillable="true" type="xsd:string"/>
    <element name="emailFrom" nillable="true" type="xsd:string"/>
    <element name="emailReplyTo" nillable="true" type="xsd:string"/>
    <element name="emailSubject" nillable="true" type="xsd:string"/>
    <element name="emailTo" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
<complexType name="FaxDeliveryOption">
  <sequence>
    <element name="faxNumber" nillable="true" type="xsd:string"/>
    <element name="faxServer" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
<complexType name="FTPDeliveryOption">
  <sequence>
    <element name="ftpServerName" nillable="true" type="xsd:string"/>
    <element name="ftpUserName" nillable="true" type="xsd:string"/>
    <element name="ftpUserPassword" nillable="true" type="xsd:string"/>
    <element name="remoteFile" nillable="true" type="xsd:string"/>
    <element name="sftpOption" type="xsd:boolean"/>
  </sequence>
</complexType>
<complexType name="LocalDeliveryOption">
  <sequence>
    <element name="destination" nillable="true" type="xsd:string"/>
  </sequence>
</complexType>
<complexType name="PrintDeliveryOption">
  <sequence>
    <element name="printNumberOfCopy" nillable="true" type="xsd:string"/>
    <element name="printRange" nillable="true" type="xsd:string"/>
    <element name="printSide" nillable="true" type="xsd:string"/>
    <element name="printTray" nillable="true" type="xsd:string"/>
    <element name="printerName" nillable="true" type="xsd:string"/>
  </sequence>

```

```

    </sequence>
  </complexType>
  <complexType name="WebDAVDeliveryOption">
    <sequence>
      <element name="deliveryAuthType" nillable="true" type="xsd:string"/>
      <element name="deliveryAuthTypeBasic" nillable="true" type="xsd:string"/>
      <element name="deliveryAuthTypeDigest" nillable="true" type="xsd:string"/>
      <element name="password" nillable="true" type="xsd:string"/>
      <element name="remoteFilePath" nillable="true" type="xsd:string"/>
      <element name="server" nillable="true" type="xsd:string"/>
      <element name="userName" nillable="true" type="xsd:string"/>
    </sequence>
  </complexType>
  <complexType name="DeliveryRequest">
    <sequence>
      <element name="emailOption" nillable="true" type="impl:EmailDeliveryOption"/>
      <element name="faxOption" nillable="true" type="impl:FaxDeliveryOption"/>
      <element name="ftpOption" nillable="true" type="impl:FTPDeliveryOption"/>
      <element name="localOption" nillable="true" type="impl:LocalDeliveryOption"/>
      <element name="printOption" nillable="true" type="impl:PrintDeliveryOption"/>
      <element name="webDAVOption" nillable="true"
type="impl:WebDAVDeliveryOption"/>
    </sequence>
  </complexType>
  <complexType name="ScheduleRequest">
    <sequence>
      <element name="deliveryRequest" nillable="true" type="impl:DeliveryRequest"/>
      <element name="endDate" nillable="true" type="xsd:dateTime"/>
      <element name="jobCalendar" nillable="true" type="xsd:string"/>
      <element name="jobLocale" nillable="true" type="xsd:string"/>
      <element name="jobTZ" nillable="true" type="xsd:string"/>
      <element name="notificationTo" nillable="true" type="xsd:string"/>
      <element name="notifyWhenFailed" type="xsd:boolean"/>
      <element name="notifyWhenSuccess" type="xsd:boolean"/>
      <element name="notifyWhenWarning" type="xsd:boolean"/>
      <element name="repeatCount" type="xsd:int"/>
      <element name="repeatInterval" type="xsd:int"/>
      <element name="reportRequest" nillable="true" type="impl:ReportRequest"/>
      <element name="saveDataOption" type="xsd:boolean"/>
      <element name="saveOutputOption" type="xsd:boolean"/>
      <element name="scheduleBurstringOption" type="xsd:boolean"/>
      <element name="schedulePublicOption" type="xsd:boolean"/>
      <element name="startDate" nillable="true" type="xsd:dateTime"/>
      <element name="useUTF8Option" type="xsd:boolean"/>
      <element name="userJobName" nillable="true" type="xsd:string"/>
    </sequence>
  </complexType>
</schema>
</wsdl:types>

  <wsdl:message name="runReportResponse">
    <wsdl:part name="runReportReturn" type="impl:ReportResponse"/>
  </wsdl:message>

  <wsdl:message name="scheduleReportResponse">
    <wsdl:part name="scheduleReportReturn" type="xsd:string"/>
  </wsdl:message>

```

```
</wsdl:message>

<wsdl:message name="getReportDefinitionRequest">

    <wsdl:part name="reportAbsolutePath" type="xsd:string"/>

    <wsdl:part name="userID" type="xsd:string"/>

    <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="getReportParametersResponse">

    <wsdl:part name="getReportParametersReturn"
type="impl:ArrayOfParamNameValue"/>

</wsdl:message>

<wsdl:message name="getReportParametersRequest">

    <wsdl:part name="reportRequest" type="impl:ReportRequest"/>

    <wsdl:part name="userID" type="xsd:string"/>

    <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="InvalidParametersException">

    <wsdl:part name="fault" type="impl:InvalidParametersException"/>

</wsdl:message>

<wsdl:message name="getFolderContentsResponse">

    <wsdl:part name="getFolderContentsReturn" type="impl:ArrayOfItemData"/>

</wsdl:message>

<wsdl:message name="OperationFailedException">

    <wsdl:part name="fault" type="impl:OperationFailedException"/>

</wsdl:message>

<wsdl:message name="AccessDeniedException">

    <wsdl:part name="fault" type="impl:AccessDeniedException"/>

</wsdl:message>

<wsdl:message name="validateLoginResponse">

    <wsdl:part name="validateLoginReturn" type="xsd:boolean"/>

</wsdl:message>

<wsdl:message name="scheduleReportRequest">
```

```
<wsdl:part name="scheduleRequest" type="impl:ScheduleRequest"/>

<wsdl:part name="userID" type="xsd:string"/>

<wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="runReportRequest">

  <wsdl:part name="reportRequest" type="impl:ReportRequest"/>

  <wsdl:part name="userID" type="xsd:string"/>

  <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="getFolderContentsRequest">

  <wsdl:part name="folderAbsolutePath" type="xsd:string"/>

  <wsdl:part name="userID" type="xsd:string"/>

  <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="getReportDefinitionResponse">

  <wsdl:part name="getReportDefinitionReturn" type="impl:ReportDefinition"/>

</wsdl:message>

<wsdl:message name="hasReportAccessRequest">

  <wsdl:part name="reportAbsolutePath" type="xsd:string"/>

  <wsdl:part name="userID" type="xsd:string"/>

  <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:message name="hasReportAccessResponse">

  <wsdl:part name="hasReportAccessReturn" type="xsd:boolean"/>

</wsdl:message>

<wsdl:message name="validateLoginRequest">

  <wsdl:part name="userID" type="xsd:string"/>

  <wsdl:part name="password" type="xsd:string"/>

</wsdl:message>

<wsdl:portType name="PublicReportService">
```

```
<wsdl:operation name="getReportParameters" parameterOrder="reportRequest
userID password">

    <wsdl:input message="impl:getReportParametersRequest"
name="getReportParametersRequest" />

    <wsdl:output message="impl:getReportParametersResponse"
name="getReportParametersResponse" />

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException" />

    <wsdl:fault message="impl:AccessDeniedException"
name="AccessDeniedException" />

    <wsdl:fault message="impl:OperationFailedException"
name="OperationFailedException" />

</wsdl:operation>

<wsdl:operation name="validateLogin" parameterOrder="userID password">

    <wsdl:input message="impl:validateLoginRequest"
name="validateLoginRequest" />

    <wsdl:output message="impl:validateLoginResponse"
name="validateLoginResponse" />

</wsdl:operation>

<wsdl:operation name="runReport" parameterOrder="reportRequest userID
password">

    <wsdl:input message="impl:runReportRequest" name="runReportRequest" />

    <wsdl:output message="impl:runReportResponse" name="runReportResponse" />

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException" />

    <wsdl:fault message="impl:AccessDeniedException"
name="AccessDeniedException" />

    <wsdl:fault message="impl:OperationFailedException"
name="OperationFailedException" />

</wsdl:operation>

<wsdl:operation name="getReportDefinition"
parameterOrder="reportAbsolutePath userID password">

    <wsdl:input message="impl:getReportDefinitionRequest"
name="getReportDefinitionRequest" />

    <wsdl:output message="impl:getReportDefinitionResponse"
name="getReportDefinitionResponse" />

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException" />
```

```
<wsdl:fault message="impl:AccessDeniedException"
name="AccessDeniedException"/>

<wsdl:fault message="impl:OperationFailedException"
name="OperationFailedException"/>

</wsdl:operation>

<wsdl:operation name="getFolderContents" parameterOrder="folderAbsolutePath
userID password">

    <wsdl:input message="impl:getFolderContentsRequest"
name="getFolderContentsRequest"/>

    <wsdl:output message="impl:getFolderContentsResponse"
name="getFolderContentsResponse"/>

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException"/>

    <wsdl:fault message="impl:AccessDeniedException"
name="AccessDeniedException"/>

    <wsdl:fault message="impl:OperationFailedException"
name="OperationFailedException"/>

</wsdl:operation>

<wsdl:operation name="scheduleReport" parameterOrder="scheduleRequest userID
password">

    <wsdl:input message="impl:scheduleReportRequest"
name="scheduleReportRequest"/>

    <wsdl:output message="impl:scheduleReportResponse"
name="scheduleReportResponse"/>

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException"/>

    <wsdl:fault message="impl:AccessDeniedException"
name="AccessDeniedException"/>

    <wsdl:fault message="impl:OperationFailedException"
name="OperationFailedException"/>

</wsdl:operation>

<wsdl:operation name="hasReportAccess" parameterOrder="reportAbsolutePath
userID password">

    <wsdl:input message="impl:hasReportAccessRequest"
name="hasReportAccessRequest"/>

    <wsdl:output message="impl:hasReportAccessResponse"
name="hasReportAccessResponse"/>

    <wsdl:fault message="impl:InvalidParametersException"
name="InvalidParametersException"/>
```



```

        </wsdl:operation>

    </wsdl:portType>

    <wsdl:binding name="PublicReportServiceSoapBinding"
type="impl:PublicReportService">

        <wsdlsoap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>

        <wsdl:operation name="getReportParameters">

            <wsdlsoap:operation soapAction=""/>

            <wsdl:input name="getReportParametersRequest">

                <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

            </wsdl:input>

            <wsdl:output name="getReportParametersResponse">

                <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

            </wsdl:output>

            <wsdl:fault name="InvalidParametersException">

                <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

            </wsdl:fault>

            <wsdl:fault name="AccessDeniedException">

                <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="AccessDeniedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

            </wsdl:fault>

            <wsdl:fault name="OperationFailedException">

                <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="OperationFailedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

```

```
</wsdl:fault>

</wsdl:operation>

<wsdl:operation name="validateLogin">

  <wsdlsoap:operation soapAction="" />

  <wsdl:input name="validateLoginRequest">

    <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

  </wsdl:input>

  <wsdl:output name="validateLoginResponse">

    <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

  </wsdl:output>

</wsdl:operation>

<wsdl:operation name="runReport">

  <wsdlsoap:operation soapAction="" />

  <wsdl:input name="runReportRequest">

    <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

  </wsdl:input>

  <wsdl:output name="runReportResponse">

    <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

  </wsdl:output>

  <wsdl:fault name="InvalidParametersException">

    <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

  </wsdl:fault>
```

```
<wsdl:fault name="AccessDeniedException">

    <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="AccessDeniedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

</wsdl:fault>

<wsdl:fault name="OperationFailedException">

    <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="OperationFailedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

</wsdl:fault>

</wsdl:operation>

<wsdl:operation name="getReportDefinition">

    <wsdlsoap:operation soapAction=""/>

    <wsdl:input name="getReportDefinitionRequest">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:input>

    <wsdl:output name="getReportDefinitionResponse">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:output>

    <wsdl:fault name="InvalidParametersException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

    <wsdl:fault name="AccessDeniedException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="AccessDeniedException"
```

```
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

</wsdl:fault>

<wsdl:fault name="OperationFailedException">

    <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="OperationFailedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

</wsdl:fault>

</wsdl:operation>

<wsdl:operation name="getFolderContents">

    <wsdlsoap:operation soapAction=""/>

    <wsdl:input name="getFolderContentsRequest">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:input>

    <wsdl:output name="getFolderContentsResponse">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:output>

    <wsdl:fault name="InvalidParametersException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

    <wsdl:fault name="AccessDeniedException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="AccessDeniedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

    <wsdl:fault name="OperationFailedException">
```

```

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="OperationFailedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

</wsdl:operation>

<wsdl:operation name="scheduleReport">

    <wsdlsoap:operation soapAction=""/>

    <wsdl:input name="scheduleReportRequest">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:input>

    <wsdl:output name="scheduleReportResponse">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:output>

    <wsdl:fault name="InvalidParametersException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

    <wsdl:fault name="AccessDeniedException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="AccessDeniedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

    </wsdl:fault>

    <wsdl:fault name="OperationFailedException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="OperationFailedException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded"/>

```

```
</wsdl:fault>

</wsdl:operation>

<wsdl:operation name="hasReportAccess">

    <wsdlsoap:operation soapAction="" />

    <wsdl:input name="hasReportAccessRequest">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

    </wsdl:input>

    <wsdl:output name="hasReportAccessResponse">

        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

    </wsdl:output>

    <wsdl:fault name="InvalidParametersException">

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
name="InvalidParametersException"
namespace="http://xmlns.oracle.com/oxp/service/PublicReportService"
use="encoded" />

    </wsdl:fault>

</wsdl:operation>

</wsdl:binding>

<wsdl:service name="PublicReportServiceService">

    <wsdl:port binding="impl:PublicReportServiceSoapBinding"
name="PublicReportService">

        <wsdlsoap:address
location="http://localhost:8080/xmlpserver/services/PublicReportService" />

    </wsdl:port>

</wsdl:service>

</wsdl:definitions>
```

New Features in BI Publisher Release 10.1.3.4

Oracle BI EE Release 10.1.3.4 introduces several new features for Oracle BI Publisher. These are outlined in the following list and explained in detail in the updated documentation for BI Publisher Release 10.1.3.4:

- [Section 28.1, "Integration with Oracle Enterprise Performance Management Workspace"](#)
- [Section 28.2, "Integration with Oracle Smart Space"](#)
- [Section 28.3, "Support for Multidimensional Data Sources"](#)
- [Section 28.4, "Enhancements to the BI Publisher Web Service"](#)
- [Section 28.5, "Support for "After Report" Triggers"](#)
- [Section 28.6, "Support for Complex Web Services as a Data Source"](#)
- [Section 28.7, "New Report Property "Disable Client Access from Analyzer for Excel""](#)
- [Section 28.8, "Enhancements to the Template Builder for Microsoft Word Add-in"](#)

28.1 Integration with Oracle Enterprise Performance Management Workspace

Oracle Enterprise Performance Management Workspace (EPM Workspace) is a component of Oracle's Hyperion Foundation Services. It is the central Web interface for users to access all Performance Management content and tools. With its ease-of-use and flexibility, EPM Workspace provides users with a "windows-on-the-Web" experience. In Release 10.1.3.4, BI Publisher can be integrated into the Oracle Enterprise Performance Management Workspace, Fusion Edition 11.1.1 release.

For more information, see the topic "Setting Up Integration with Oracle Enterprise Performance Management Workspace," in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

28.2 Integration with Oracle Smart Space

Oracle Smart Space, Fusion Edition is a personalized information delivery solution for Business Intelligence and Enterprise Performance Management that uses configurable desktop gadgets to deliver BI and EPM to every user in the enterprise, utilizing the Windows desktop. In addition, Oracle Smart Space includes a development toolkit for

creating additional gadgets that employs common development languages and methodologies and a secure instant messaging system for shared decision making.

You can use gadgets from Oracle Smart Space, Fusion Edition 11.1.1 release to access and interact with the Release 10.1.3.4 BI Publisher reports repository. For example, you can launch BI Publisher reports in the Oracle Smart Space Content Viewer and add BI Publisher reports to a Smart Book gadget and a Favorites gadget. You can use Oracle Smart Space Collaborator to share BI Publisher content with other Oracle Smart Space users.

For more information, see the topic: "Using Oracle BI Publisher in Oracle Smart Space, Fusion Edition," in *Oracle Business Intelligence Publisher Report Designer's Guide*.

This integration also enables you to set up a link to the Oracle Smart Space client download page that is accessible to all BI Publisher users. For more information see the topic: "Setting Up Integration with the Oracle Smart Space Client Download Page," in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

28.3 Support for Multidimensional Data Sources

BI Publisher now enables you to connect to databases that are configured for online analytical processing (OLAP) and submit an MDX query to return the report XML. In this release, BI Publisher supports connections to the following multidimensional data sources:

- Oracle's Hyperion Essbase
- Microsoft SQL Server 2000 Analysis Service
- Microsoft SQL Server 2005 Analysis Service
- SAP Business Information Warehouse (SAP BW)

For more information, see the topic: "Setting Up a Connection to an OLAP Data Source" in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

28.4 Enhancements to the BI Publisher Web Service

In Release 10.1.3.4, the BI Publisher Web service `PublicReportService` has been expanded to support several more operations. These new operations include:

- Operations to run reports
 - `uploadReportDataChunk`
 - `downloadReportDataChunk`
- Operations to Manage Scheduled Reports and History of Scheduled Reports
 - `suspendScheduledReport`
 - `resumeScheduledReport`
 - `deleteScheduledReport`
 - `getScheduledReport`
 - `getScheduledReportInfo`
 - `getScheduledReportHistoryInfo`
 - `deleteScheduledReportHistory`
 - `getScheduledReportStatus`

- Operations to Create and Manage Reports
 - createReport
 - deleteReport
 - updateReportDefinition
 - createReportFolder
 - uploadTemplateForReport
 - removeTemplateForReport
 - uploadReport
 - downloadReport
- Operations about the Report Server
 - getSecurityModel
 - getBIPHTTPSessionInterval

For more information about these new Web service operations, see the chapter: "Using the BI Publisher Web Service APIs" in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

28.5 Support for "After Report" Triggers

BI Publisher now provides the ability to set up an HTTP notification as part of the report schedule. When the report completes, BI Publisher sends a notification to a designated URL, along with the report job information. Your remote application can use this information with the BI Publisher APIs and Web services to access the job details, including the report output and XML data to enable third-party postprocessing of the report.

For more information, see the topic: "Setting Up an After Report Trigger" in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*.

28.6 Support for Complex Web Services as a Data Source

In previous releases, BI Publisher could accept only simple Web services that returned simple data types, such as string and Boolean. Starting with Release 10.1.3.4, BI Publisher can accept Web services that return complex data types. The Web service data source user interface has been enhanced so that when you provide the WSDL, BI Publisher presents you with the list of Web services that are available from the WSDL, the methods available within the selected service, and the required parameters for the selected method.

For more information see the topic: "Defining a Web Service Data Set Type" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

28.7 New Report Property "Disable Client Access from Analyzer for Excel"

In this release BI Publisher introduces a new method for downloading report data to the Excel Analyzer. The new property "Disable Client Access from Analyzer for Excel" enables you to control the method used to download data to Microsoft Excel at the report level. Turning this property on has the following effects:

- Your report data downloads faster and large data sets are handled more efficiently.
- You do not have to enable macros when you download the data.
- You can enable your own custom macros in Microsoft Excel.
- You cannot log in or connect to the BI Publisher server from your Microsoft Excel session. Therefore you cannot upload a template directly from Excel, nor can you update the report parameters or apply a new template.

For more information about this property, see the topic "Using the Report Editor" in *Oracle Business Intelligence Publisher Report Designer's Guide*.

28.8 Enhancements to the Template Builder for Microsoft Word Add-in

Release 10.1.3.4 of BI Publisher introduces the following enhancements for the Template Builder for Microsoft Word:

- **Page Break Support**
 You can now insert a page break in the report layout when setting properties for a repeating group. When you insert a repeating group, the Properties dialog now includes the following options:
 - No break
 - Page break
 - Section break
 When you select page break, BI Publisher inserts a page break between each occurrence of the element selected in the "For each" field.
 The break options are also available within the Table Wizard when defining groups for a table.
 For more information, see the topic: "Inserting a Repeating Group" in the Template Builder online help.
- **Support for Additional Chart Types**
 The following chart types are now supported by the Template Builder chart wizard:
 - Combination Charts
 - Scatter Plot
 - Bubble Chart
 - Stock Charts
 For more information, see the topic: "Inserting a Chart" in the Template Builder online help.
- **More Properties Supported for Charts**
 This release includes expanded support for setting chart properties within the Chart Wizard. You can now set properties for the following:
 - Fonts - you can define the font size, family, weight, color (and effect) for the x-axis and y-axis, the chart legend, and chart labels.
 - Colors - you can now specify the color for bars and lines, the chart plot area and legend, the background, and the border of the plot area and legend.

- Chart legend properties - you can now define the look and feel of the legend including the legend background and border color and the legend font family and font size. You can also define the position of the legend as follows: bottom, top, left, right.
- Support for Multiple Measures in Charts

The chart builder now supports multiple measures. You can also specify the name displayed in the legend for each measure.
- Enhanced Crosstab Features

The crosstab builder has been enhanced to support the following:

 - Total Options - previously, only the sum was displayed and it was always displayed. Now you can define whether to show totals or totals based on sum, count, min, max or average.
 - Number Formatting - you can now define the number formatting for the crosstab measures within the crosstab builder.
 - Create and Omit Subtotal and Totals - the crosstab builder now enables you to choose whether to display total and subtotal rows or columns.
 - Row Layout Options - you can now select whether you want the row groups to be displayed inline (as in a financial report) or not inline (as in Microsoft Excel).

New Features in BI Publisher Release 10.1.3.4.1

Oracle BI EE Release 10.1.3.4.1 introduces several new features for Oracle BI Publisher. These are outlined in the following list and explained in detail in *Oracle Business Intelligence Publisher New Features Guide*:

- Support for Oracle WebLogic Server 10.3
- Extended Support for Single Sign-On Providers
- Automatic Refresh of LDAP Cache
- Support for Siebel CRM Security
- Support for Secure File Transfer Protocol (SFTP) for Burst Reports
- Data Model Enhancements
- Scheduler Support for JNDI Connection Pools
- Support for Expressions to Calculate Date Parameters
- RTF Template Enhancements
- Remove Logos and Links from the BI Publisher Header
- Disable Access to Guest Page for Unauthorized Users
- Updates to the BI Publisher Web Service

New Features in BI Publisher Release 10.1.3.4.2

Oracle BI EE Release 10.1.3.4.2 introduces several new features for Oracle BI Publisher. These are outlined in the following list and explained in detail in *Oracle Business Intelligence Publisher New Features Guide*:

- Display Scheduler Job Times in the User's Preferred Time Zone
- Attachment Name Field for E-mail Deliveries
- PDF Output Property "PDF Navigation Panels Initial View"
- Support for Safe Divide Function
- Support for Hyperlinking Across Templates
- Support for "Comb of Characters" Option for Form Fields in PDF Templates
- Support for JDBC Triggers
- Integrating with Microsoft Active Directory
- Using LDAP Attribute Values as Bind Variables in Data Queries
- Support for IBM WebSphere 7.x
- Support for Excel Templates

Part III

New Features for Oracle Business Intelligence for Microsoft Office

Part III contains the following chapters:

- [Chapter 31, "Installing Oracle BI for Microsoft Office in Release 10.1.3.3"](#)
- [Chapter 32, "Microsoft Office Support in Release 10.1.3.3"](#)
- [Chapter 33, "New Features in Oracle BI for Microsoft Office Release 10.1.3.3.1"](#)
- [Chapter 34, "New Features in Oracle BI for Microsoft Office Release 10.1.3.3.3"](#)
- [Chapter 35, "New Features in Oracle BI for Microsoft Office Release 10.1.3.4"](#)
- [Chapter 36, "New Features in Oracle BI for Microsoft Office Release 10.1.3.4.1"](#)
- [Chapter 37, "New Features in Oracle BI for Microsoft Office Release 10.1.3.4.2"](#)

Installing Oracle BI for Microsoft Office in Release 10.1.3.3

Oracle Business Intelligence for Microsoft Office includes the Oracle BI Office server and the Oracle Business Intelligence Add-in for Microsoft Office client. The client includes both the Oracle BI Add-In for Microsoft Excel and the Oracle BI Add-in for Microsoft PowerPoint. This chapter describes how to install and configure the add-in and the BI Office Server.

This chapter includes the following sections:

- [Section 31.1, "System Requirements"](#)
- [Section 31.2, "Installing the BI Office Server"](#)
- [Section 31.3, "Deploying the BI Office Server on J2EE Application Servers"](#)
- [Section 31.4, "Installing the Oracle BI Add-In for Microsoft Office"](#)
- [Section 31.5, "Setting Properties in the bioffice.xml Configuration File"](#)
- [Section 31.6, "Setting Up SSL-Enabled OC4J for Oracle BI for Microsoft Office"](#)
- [Section 31.7, "Setting Up an SSO-Enabled Presentation Server for Releases Before 10.1.3.4"](#)
- [Section 31.8, "Setting Up an SSO-Enabled Presentation Server for Releases 10.1.3.4 and Later"](#)
- [Section 31.9, "Setting Up an NTLM-Enabled Presentation Server for Releases 10.1.3.4 and Later"](#)

31.1 System Requirements

Following are the required components for Oracle BI for Microsoft Office. For the detailed list of system requirements for Oracle BI EE and for updates to this list, see *System Requirements and Supported Platforms for Oracle Business Intelligence Suite Enterprise Edition, Version 10.1.3.4*.

31.1.1 Supported Client Operating Systems

- Windows XP
- Windows Vista

31.1.2 Supported Versions of Microsoft Office

- Microsoft Office 2003

- Microsoft Office 2007

31.1.3 Supported Application Servers

- Oracle Application Server (OC4J/HTTP) 10.1.3.1 on Linux
- Oracle Application Server 10.1.3.3
- Apache Tomcat 5.5x on Linux
- BEA Weblogic 9.x
- IBM Websphere Application Server 5.1 on Windows
- IBM Websphere Application Server 6.0, 6.1
- Sun Java System Web Server 6.1, 7.0

31.2 Installing the BI Office Server

The Oracle BI EE Release 10.1.3.3 (and later) installer installs the BI Office Server. The installer performs the following tasks:

- Deploys the biooffice.ear file.
- Initializes values in the configuration file for the BI Office Server.
- Copies the client OracleBIOffice.exe file to the location where Presentation Services is running.
- Updates the instanceconfig.xml file with the location of the client OracleBIOffice.exe.
- Backs up the BI Office Server configuration file, biooffice.xml, to a subdirectory under <OracleBI_Home>/Install_Backup/<upgrade_timestamp>.

Note: During an upgrade installation, this file is used to retain the existing configuration settings.

31.2.1 Deployment of the biooffice.ear

The Oracle BI EE 10.1.3.3 (and later) installer installs the Oracle BI Office Server into the J2EE container with the Presentation Services Plug-in.

- The basic Oracle BI EE installation option deploys the biooffice.ear to the standalone OC4J home.
- The advanced Oracle BI EE installation option creates a new OC4J instance called "biooffice" in Oracle Application Server and deploys the biooffice.ear to this new OC4J instance.

31.2.2 Initialization of the Server Configuration File

The BI Office Server configuration file is biooffice.xml. The location of the file depends on the installation type. For example:

- In an Oracle BI EE basic installation, it is installed under <OracleBI_Home>/oc4j_bi/j2ee/home/applications/biooffice/biooffice/WEB-INF/biooffice.xml
- In an Oracle BI EE advanced installation, it is installed under ORACLE_HOME/j2ee/bibiooffice/applications/biooffice/biooffice/WEB-INF/biooffice.xml

- In an Oracle BI EE installation with IBM WebSphere 5.1 (Windows), it is installed under C:\Program Files\WebSphere\AppServer\installedApps\<server name>\biooffice.ear\biooffice.war
- In an Oracle BI EE installation with Apache Tomcat 5.5 (Windows), it is installed under C:\Program Files\<Apache_Home>\Tomcat 5.5\webapps\biooffice
- In other non-Oracle Application Servers, it is installed under the biooffice application directory, parallel to the Oracle BI analytics application.

The Oracle BI EE installer configures the BI Office Server to point to the Oracle BI Presentation Services server by setting the property "SawBaseURL" in the biooffice.xml file. To configure the BI Office Server to point to a different Presentation Services server, modify this property with the new Presentation Services URL. For more details about other properties in the biooffice.xml file, see [Section 31.5, "Setting Properties in the biooffice.xml Configuration File."](#)

31.2.3 Location of the Installed Client OracleBIOffice.exe File

The installer copies the client/OracleBIOffice.exe file to the location where Presentation Services is running and updates the instanceconfig.xml file with this location. This makes the client install executable available for download from the More Products menu in Oracle BI Interactive Dashboards and Oracle BI Answers.

For example, in an OC4J install, the OracleBIOffice.exe is installed to <OracleBI_Home>/oc4j_bi/j2ee/home/applications/analytics/analytics/client/OracleBIOffice.exe.

In an Oracle Application Server install, the OracleBIOffice.exe is installed to ORACLE_HOME/j2ee/bianalytics/applications/analytics/analytics/client/OracleBIOffice.exe.

The installer modifies the instanceconfig.xml (OracleBIData/web/config/instanceconfig.xml) by adding the new element <BIforOfficeURL> and initializing the property to point to the location of the client download. For example:

```
<BIforOfficeURL>client/OracleBIOffice.exe</BIforOfficeURL>
```

31.3 Deploying the BI Office Server on J2EE Application Servers

If you deployed Oracle BI EE on a non-Oracle J2EE Application Server, then you must manually deploy the biooffice.ear or biooffice.war file to use the BI Office Server. It is not included in the analytics.ear file. The biooffice application must be deployed to the same container as the analytics application.

To deploy the BI Office Server:

1. From a browser, open the Web application deployment page for the J2EE application server.
2. Locate the biooffice.ear or biooffice.war file under OracleBI/office/server.
3. Using the guidelines for your application server, deploy the biooffice.ear or biooffice.war file as an application named "biooffice" to your J2EE application server environment.

To make the client executable files available to Presentation Services (required for Release 10.1.3.3 only):

Note: Do not perform this step if you are installing Oracle BI EE Release 10.1.3.3.1 or later. In the 10.1.3.3.1 and later releases, the installer automatically places the client executable file in the deployed analytics/analytics/client directory.

Copy the <OracleBI_Home>/office/client directory to the deployed analytics/analytics/client directory.

To update configuration files:

1. Update the instanceconfig.xml to point to the location of the OracleBIOffice.exe file.

- a. Locate the instanceconfig.xml file in the directory <OracleBI_Home>/web/config.

- b. Add the following entry to the file:

```
<BIforOfficeURL>client/OracleBIOffice.exe</BIforOfficeURL>
```

Note: For more information on modifying instanceconfig.xml, refer to *Oracle Business Intelligence Presentation Services Administration Guide*.

2. Update the bioffice.xml file to point to the Presentation Services server.

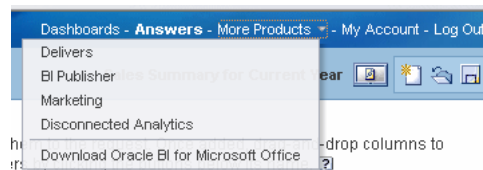
Update the property <SawBaseURL> with the host name and port of the Presentation Services server. See [Section 31.5, "Setting Properties in the bioffice.xml Configuration File"](#) for the file location and more information.

31.4 Installing the Oracle BI Add-In for Microsoft Office

After installing Oracle BI EE, you can download the Oracle BI Add-In for Microsoft Office from Oracle BI Answers and Interactive Dashboards.

To install the add-in:

1. Log in to Oracle BI Answers or Interactive Dashboards with your Oracle BI EE user credentials.
2. Select the **More Products** link and then select **Download Oracle BI for Microsoft Office**.



3. A dialog prompts you to run or save OracleBIOffice.exe. Save the file to a local directory.
4. Navigate to the saved location. Close all Microsoft Office applications. Double-click OracleBIOffice.exe to launch the installer.

Microsoft .Net Framework 2.0 is required for the add-in. If it is not installed on your computer, then you are prompted to exit and install Microsoft .Net Framework 2.0 from the Microsoft Web site.

Note: In Oracle BI EE Release 10.1.3.3, Microsoft .Net Framework 2.0 was included with the add-in's installer. If you are installing this version, then you are prompted to install Microsoft .Net Framework 2.0 without being required to exit.

In addition, if the following prerequisites do not exist on your computer, then you are prompted to install them:

- Office2003-kb907417sfxcab-ENU_Patch
- Shared Add-In Extensibility Microsoft .Net 2.0
- Shared Add-In Support Update Microsoft .Net 2.0

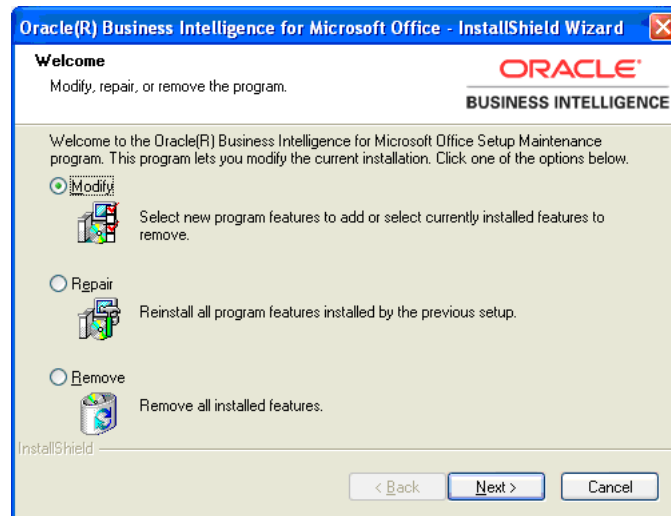
After installing the prerequisites, you might be prompted to restart the computer.

After restarting, navigate back to the location of the OracleBIOffice.exe file and double-click to reinitiate the install.

5. From the Welcome screen, select **Next**.
6. In the Customer Information screen, enter your User Name and Company Name, and select the desired sharing option.
7. On the Setup Type screen, choose one of the following:
 - **Typical** — Performs installation of both add-ins to C:\Program Files\Oracle\BIOffice
 - **Custom** — Enables you to select the install directory, and choose which components to install.
8. After all components are installed, the InstallShield Wizard Complete dialog is displayed. Click **Finish**.

31.4.1 Performing a Maintenance or Upgrade Install

If you have an existing installation on your computer, then you are prompted to Modify, Repair or Remove the existing installation.



Modify: Prompts you to select the specific add-in to install.

Repair: Uses a cached copy of the original install.

Remove: Removes the add-ins.

31.4.2 Configuring the Client

When the software is downloaded and installed, each user must enter the connection information in the Oracle BI Add-in for Microsoft Office Preferences dialog on the client computer to enable connection to Presentation Services.

Note: The Preferences dialog was updated in Releases 10.1.3.3.1 and 10.1.3.4. If you are on Release 10.1.3.3, then the Application Name field is not available. In Release 10.1.3.4, the following field names in the dialog have been updated:

- Name is updated to Server Name
 - Office Server is updated to BI Office Server
 - Application is updated to Application Name
-

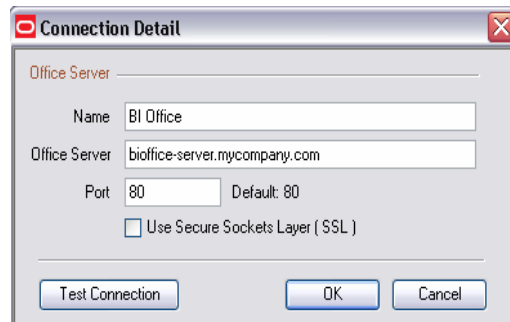
To add connection information to the client computer:

1. Open Microsoft Excel or Microsoft PowerPoint. (The connection information is shared, therefore you can enter it through either application and it is available to both.)
2. From the Oracle BI menu, select **Preferences**.
3. On the Connections tab, select **New**.
4. Enter the following fields:
 - a. **Server Name** for the connection
 - b. **BI Office Server** — The URL for the Oracle BI Office Server (for example: biooffice-server.mycompany.com)
 - c. **Application Name** - (this field was added in release 10.1.3.3.1) enter the Application Name that you defined for the BI Office server when you deployed the BI Office Server application to its J2EE container. The name

defaults to "biooffice", but if you specified another name when you deployed the application, then enter that name in this field.

Note: The value entered in the Application Name field is appended to the values for BI Office Server URL and Port to construct the URL that the client uses to connect to the BI Office server application. For example: `http://biooffice-server.mycompany.com:80/biooffice`

- d. **Port** for the BI Office Server. The default is 80.



Note: If your environment is SSL enabled, then see [Section 31.6, "Setting Up SSL-Enabled OC4J for Oracle BI for Microsoft Office"](#) for the appropriate setup steps.

If you require security for the data that is passed between the client and server, then host Oracle BI EE on a secured HTTPS server.

5. Click **Test Connection** to test the connection between the add-in and the BI Office Server. (Note that this does not test the connection between the Oracle BI Office Server and Presentation Services.)

31.5 Setting Properties in the biooffice.xml Configuration File

This section describes the properties that you can configure in the biooffice.xml file. For file location see [Section 31.2.2, "Initialization of the Server Configuration File"](#).

Following is a sample biooffice.xml file. The sample file is from Release 10.1.3.4. If you are using a release before 10.1.3.4, then the file does not contain the following properties:

- SawWebURLforSSO
- SawSSOImpersonator
- SawSSOImpersonatorPassword
- OfficeCompressResult
- OfficeCompressResultMinSize

```
<biooffice>
  <!-- log -->
  <!-- LogDir. Default is [04CJ
dir]\j2ee\home\applications\biooffice\biooffice\WEB-INF\log -->
  <!-- <property name="LogDir">D:\BIOffice\Server\log\</property> -->
  <!-- LogLevel Never = 1; Error = 2; Warning = 3; Msg = 4; Debug = 5; -->
```

```

    <property name="LogLevel" type="int">3</property>
    <!-- saw -->
    <property name="SawBaseURL">http://localhost/analytics/saw.dll</property>
    <!-- Does SAW use SSO (Single Sign-On): yes = 1; no = 0; -->
    <property name="SawUseSSO" type="int">0</property>
    <!-- In SSO/NTLM case, sawBaseURL is for web service only,
    sawWebURLforSSO is for external web request. In none SSO/NTLM
    case, sawBaseURL is for both web service and external web request.
    -->
    - <!-- <property
name="SawWebURLforSSO">http://localhost/analytics/saw.dll</property>
    -->
    - <!-- <property name="SawSSOImpersonator"></property>
    -->
    - <!-- <property name="SawSSOImpersonatorPassword"></property>
    -->
    <!-- Specify the maximum number of rows to be returned by SAW executeXMLQuery
    or fetchNext method. -->
    <property name="SawMaxRowsPerPage" type="int">5000</property>
    <!-- Parse hyperlink column? Yes = 1; No = 0; -->
    <property name="SawParseHyperLink" type="int">1</property>
    <!-- Parse percentage column: Yes = 1; No = 0; -->
    <property name="SawParsePercentageColumn" type="int">1</property>
    <!-- Fetch fresh data: Yes = 1; No = 0; -->
    <!-- Turn on makes SAW re-submits the query to refresh data, in stead of
    reading data from cache. -->
    <property name="SawFetchFreshData" type="int">0</property>
    <!-- Office
    -->
    - <!-- Compress result sent to Office client? Yes = 1; No = 0;
    -->
    <property name="OfficeCompressResult" type="int">1</property>
    - <!-- If compress Office result, specify the minimum size (in bytes) to trigger
    the compression. -->
    <property name="OfficeCompressResultMinSize" type="int">16384</property>
</biooffice>

```

The following table describes the biooffice.xml properties:

Property Name	Valid Values	Description
LogDir	N/A	Enter the path to the directory for the BI Office Server to write the log file. Default is: <O4CJ dir>\j2ee\home\applications\biooffice\biooffice\WEB-INF\log
LogLevel	1 = Never 2 = Error 3 = Warning (Default) 4 = Message 5 = Debug	Set the level of information you want to be written to the log file. Note that as the log level value increases, performance is impacted. Therefore do not set this property to debug (5) unless you are troubleshooting an issue.

Property Name	Valid Values	Description
SawBaseURL	<p>http://<HOSTNAME>:<PORT>/analytics/saw.dll</p> <p>or</p> <p>https://<HOSTNAME>:<PORT>/analytics/saw.dll</p>	<p>Enter the URL for Presentation Services. This property is set initially by the BI EE installer.</p> <p>Important If SSO or NTLM is enabled, then enter the URL for the protected analytics servlet that you deployed when configuring Oracle BI for Microsoft Office to integrate with the SSO-enabled BI Server. The URL that is specified for this property is used for Web service requests between Oracle BI for Microsoft Office and Presentation Services. For information, see the appropriate topic for your system: Section 31.7, "Setting Up an SSO-Enabled Presentation Server for Releases Before 10.1.3.4," Section 31.8, "Setting Up an SSO-Enabled Presentation Server for Releases 10.1.3.4 and Later," or Section 31.9, "Setting Up an NTLM-Enabled Presentation Server for Releases 10.1.3.4 and Later."</p> <p>If SSO or NTLM is not enabled, then SawBaseURL is the only URL that is required to access BI Presentation Services.</p>
SawUseSSO	<p>0 = No (Default)</p> <p>1 = Yes</p>	<p>For releases before 10.1.3.4: Default is 0. Set this property to 1 if your BI Server implementation is SSO-enabled. In an SSO-enabled implementation, the add-in features "Insert as Flash," "Insert as Image," and "Edit View" are not supported. Setting this option to "1" suppresses these options from the add-in menu.</p> <p>For versions 10.1.3.4 and later: Set this property to 1 if the Oracle BI EE implementation is enabled for SSO or NTLM.</p>
SawWebURLforSSO	<p>http://<HOSTNAME>:<PORT>/analytics/saw.dll</p> <p>or</p> <p>https://<HOSTNAME>:<PORT>/analytics/saw.dll</p>	<p>Added in Release 10.1.3.4</p> <p>When SSO or NTLM is enabled, use this property to enter the public URL that allows external users to access the Oracle BI Answers page.</p>
SawSSOImpersonator	N/A	<p>Added in Release 10.1.3.4</p> <p>When SSO or NTLM is enabled for the BI Server, use this property to enter the Impersonator user that you defined for the BI Server.</p>

Property Name	Valid Values	Description
SawSSOImpersonatorPassword	N/A	Added in Release 10.1.3.4 When SSO or NTLM is enabled for the BI Server, use this property to enter the Impersonator user password that you defined for the BI Server.
SawMaxRowsPerPage	N/A	Default setting is 5000. Specifies the maximum number of rows to be returned by SAW executeXMLQuery or fetchNext method. Increase this setting in a multiple-user environment.
SawParseHyperLink	0 = Off 1 = On (Default)	When set to 1, the BI Office Server parses the HTML content in the results set and converts the content to HTML display format in Excel. When set to 0, the BI Office Server inserts the hyperlink text as is; that is, as text with markup tags. If the data does not include any hyperlink text, then set this property to 0 for better performance.
SawParsePercentageColumn	0 = Off 1 = On (Default)	In Answers, if the Column Properties dialog for a given column is set up as follows: On the Data Format tab, "Override Default Data Format" is checked and "Treat Numbers as Percentage" is selected; then setting this property to 1 instructs the BI Office Server to treat the number as a percentage (divide by 100). If the user formats the data as a percentage in Excel, then the data displays correctly. If the property is set to 0, then no percentage processing is done by the BI Office Server. For more information on setting Column Properties in Answers, see <i>Oracle Business Intelligence Answers, Delivers, and Interactive Dashboards User Guide</i> .
SawFetchFreshData	0 = Off (Default) 1 = On	Default is 0. Set this property to 1 to resubmit the query to refresh data. When set to 0, the data is read from the cache.

Property Name	Valid Values	Description
OfficeCompressResult	0 = No 1 = Yes (Default)	<p>Added in Release 10.1.3.4</p> <p>This property is turned on by default to compress the Answers result set on the BI Office Server before being sent to the add-in. Define the minimum file size that triggers the compression in the property "OfficeCompressResultMinSize".</p> <p>Guidelines for setting this property: If the results are expected to be large, (more than 16KB or 32KB), then turn compression on. There is a performance overhead associated with compressing the result set on the server side, and then decompressing it on the client side. The benefit of compression is a significant reduction in the size of the data sent from the server to the client; because the results are usually in the form of XML, which is verbose, compression can shrink the result set by as much as 90%. The cost versus benefit can vary for your environment, therefore some trial and error may be required before achieving the optimal setting.</p>
OfficeCompressResultMinSize	Express as an integer in bytes. Default value is 16384.	<p>Added in Release 10.1.3.4</p> <p>If the property "OfficeCompressResult" is turned on, then enter the minimum size file in kilobytes that triggers compression. See the description for "OfficeCompressResult" for guidelines on setting these properties.</p>

31.6 Setting Up SSL-Enabled OC4J for Oracle BI for Microsoft Office

Oracle Containers for J2EE supports Secure Sockets Layer (SSL) communication. For information on setting up SSL in an OC4J environment, see *Oracle Containers for J2EE Security Guide 10g (10.1.3.1.0)*.

Assuming the Oracle BI Server is installed under C:\OracleBI, the following instructions describe how to set up SSL in a standalone OC4J environment on Windows.

Note: In the Oracle BI EE installation, you find the keytool utility in this location: C:\OracleBI\uninstall_jvm\bin\keytool.exe.

1. Create a keystore with an RSA private/public key pair using the keytool utility.

Example:

This example generates a keystore to reside in a file named mykeystore.jks, which has a password of 123456, using the RSA key pair generation algorithm.

```
cd C:\OracleBI\oc4j_bi\j2ee\home\config
keytool -genkey -keyalg RSA -keystore mykeystore.jks -storepass 123456
```

The keytool prompts you for additional information, as follows:

What is your first and last name?

[Unknown]: Test User

What is the name of your organizational unit?

[Unknown]: Support

What is the name of your organization?

[Unknown]: Oracle

What is the name of your City or Locality?

[Unknown]: Redwood Shores

What is the name of your State or Province?

[Unknown]: CA

What is the two-letter country code for this unit?

[Unknown]: US

Is <CN=Test User, OU=Support, O=Oracle, L=Redwood Shores, ST=CA, C=US> correct?

[no]: yes

Enter key password for <mykey>

(RETURN if same as keystore password):

Always press RETURN for the key password. In OC4J 10.1.3.x implementations, the keystore password must be the same as the key entry password.

The mykeystore file is created in the current directory. The default alias of the key is mykey.

2. If you do not have a secure-web-site.xml file, then create one in the following location: C:\OracleBI\j2ee\home\config\secure-web-site.xml. You can start by copying content from default-web-site.xml.
3. Update secure-web-site.xml with the following elements:
 - a. Update the <web-site> element to add secure="true" and to set the port value to some available port. (For example, port="4443". To use the default of 4443, you must be a super user.) For standalone OC4J, use HTTP protocol, which is the default setting. (The setting protocol="http" in combination with secure="true" results in HTTPS being used.)


```
<web-site port="4443" secure="true" protocol="http" display-name="Default
OracleAS Containers for J2EE Web Site" >
...
</web-site>
```
 - b. Add the following under the <web-site> element to define the keystore and password.


```
<ssl-config keystore="./mykeystore.jks" keystore-password="123456" />
```

Note that the relative path and file name extensions are used, for example: ./mykeystore.jks.
 - c. Ensure that the server.xml points to the secure-web-site.xml file.

As necessary, uncomment or add the following line in server.xml:

```
<web-site path="./secure-web-site.xml" />
```

4. Restart OC4J to initialize the secure-web-site.xml file additions.
5. Test the SSL port by accessing the site in a browser on the SSL port. For example: <https://bieeserver.mycompany.com:4443/em>
6. Import the certificate into the Oracle BI JDK and export the self-signed certificate.

- a. `cd JAVA_HOME\jre\lib\security`

- b. Use the keytool utility to export a certificate from your keystore to mykeystore.cer.

```
keytool -export -keystore C:\OracleBI\oc4j_\
bi\j2ee\home\config\mykeystore.jks -storepass 123456 -alias mykey -rfc
-file mykeystore.cer
```

- c. Use the keytool utility to import the mykeystore.cer to replace the Oracle BI JDK systemwide keystore cacerts.

```
keytool -import -keystore cacerts -storepass changeit -alias mykeycert
-file mykeystore.cer
```

7. Update SawBaseURL in the BI Office Server configuration file biooffice.xml as follows:

```
<property name="SawBaseURL">https://
/biooffice-server.mycompany.com:4443/analytics/saw.dll</property>
```

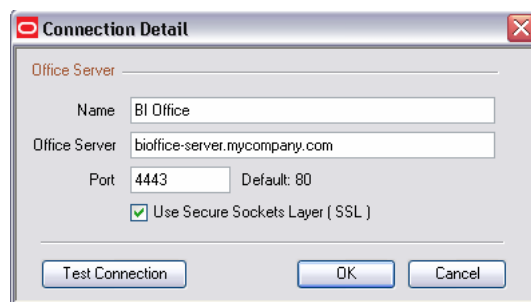
8. Now you can access this SSL-enabled OC4J and Oracle BI EE from the Oracle BI Add-In for Microsoft Office. In the Add-In, create a connection as follows:

Name: BI Office

Office Server: biooffice-server.mycompany.com

Port: 4443

Use Secure Sockets Layer (SSL): selected



Tip: If you can connect to SSL successfully in the browser, yet the connection fails in the client, then confirm that the certificate was exported and imported properly (see Step 6).

31.7 Setting Up an SSO-Enabled Presentation Server for Releases Before 10.1.3.4

Oracle BI for Microsoft Office currently cannot be fully integrated with SSO. If the Oracle BI EE implementation is SSO-enabled, then users can use their SSO credentials from the Microsoft Excel or PowerPoint add-in to log in. However, the following options from the add-in's Insert menu are not supported:

- Edit View
- Insert as Image
- Insert as Flash

To enable the integration of Oracle BI for Microsoft Office with SSO-enabled Oracle BI EE, you must deploy a separate Presentation Services Plug-in (analytics.ear) and grant trusted IP address access privileges to the new Presentation Services Plug-in from the IP address of the BI Office Server. This process enables Web service requests from the IP address of the trusted BI Office Server to be handled by the Presentation Services Plug-in without going through SSO authentication. You then must configure the BI Office Server to send Web service requests to this Presentation Services Plug-in that bypasses SSO.

Depending on the SSO server type, the configuration might differ. Following is an example of how to integrate Oracle BI for Microsoft Office with Oracle SSO-enabled Oracle BI EE:

1. On the same computer where the Presentation Services Plug-in has been deployed, deploy another Presentation Services Plug-in using the file analytics.ear:

Locate analytics.ear in the directory OracleBI_HOME/web.

2. Name the new Plug-in "analyticsSOAP".

Make the same modifications to the web.xml file for this analyticsSOAP servlet that were made to the web.xml file for the default "analytics" servlet. For more information on updating the web.xml file, see the topic "Configuring BI Presentation Services Plug-in" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

3. Make the following modification to the file mod_osso.conf to open analyticsSOAP to requests from the BI Office server:

(The mod_osso.conf is located in the directory Oracle_HOME/Apache/Apache/conf.)

```
<Location /analyticsSOAP>
  Require valid-user
  AuthType Basic
  Allow from <IP address of BI Office server>
  Satisfy Any
</Location>
where
```

<IP address of BI Office Server> is the IP address of the BI Office Server, for example: 10.155.224.7

4. Locate the BI Office Server configuration file (biooffice.xml) in the following directory:

OAS_HOME\j2ee\home\applications\biooffice\biooffice\WEB-INF

5. Update the following properties:

- "SawBaseURL" to point to the new analyticsSOAP deployed in the previous steps
- "SawUseSSO" to indicate that you have enabled SSO for Oracle BI for Microsoft Office and Presentation Services. Set the value to "1". This setting suppresses the unsupported options from the client add-in insert menu.

Example:

```
<property
name="SawBaseURL">http://localhost:port/analyticsSOAP/saw.dll</property>
<!-- Does SAW use SSO (Single Sign-On): yes = 1; no = 0; -->
<property name="SawUseSSO" type="int">1</property>
```

For more information on the configuration file, see [Section 31.5, "Setting Properties in the biooffice.xml Configuration File"](#).

31.8 Setting Up an SSO-Enabled Presentation Server for Releases 10.1.3.4 and Later

Oracle BI for Microsoft Office currently cannot be fully integrated with SSO. If the Oracle BI EE implementation is SSO-enabled, users can use their SSO credentials from the Microsoft Excel or PowerPoint add-in to log in.

To enable the integration of Oracle BI for Microsoft Office with SSO-enabled Oracle BI EE, you must deploy a separate Presentation Services Plug-in (analytics.ear) and grant trusted IP address access privileges to the new Presentation Services Plug-in from the IP address of the BI Office Server. This process enables Web service requests from the IP address of the trusted BI Office Server to be handled by the Presentation Services Plug-in without going through SSO authentication. You then must configure the BI Office Server to send Web service requests to this Presentation Services Plug-in that bypasses SSO.

Note: There is an alternative method that you can use that does not require the deployment of a second Presentation Services Plug-in. See the [Section 31.8.3, "Alternative Configuration Method."](#)

To perform the Edit View function in the add-in, you must configure the BI Office Server to route these requests to the public URL for Presentation Services. The user is challenged for credentials to sign in to Oracle BI EE to launch the Answers page for editing.

31.8.1 Prerequisites

Following are the prerequisites for configuring Oracle BI for Microsoft Office with an SSO-enabled Oracle BI Server:

1. Install Oracle BI EE, Release 10.1.3.4. For information, see *Oracle Business Intelligence Infrastructure Installation and Configuration Guide*.
2. Configure the Oracle BI Server to enable Single Sign-on. See the chapter: "Enabling Oracle Single Sign-on For Oracle Business Intelligence" in *Oracle Business Intelligence Enterprise Edition Deployment Guide* for information.

When enabling SSO for the Oracle BI Server, you are required to set up an Impersonator user. You need this Impersonator user name and password to

complete the configuration of the BI Office Server. For more information, see the topic "Creating the Oracle BI Server Impersonator User" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

31.8.2 Integrating Oracle BI for Microsoft Office with SSO-Enabled Oracle BI EE

Depending on the SSO server type, the configuration might differ. Following is an example of how to integrate Oracle BI for Microsoft Office with Oracle SSO-enabled Oracle BI EE:

1. On the same computer where the Presentation Services Plug-in has been deployed, deploy another Presentation Services Plug-in using the file `analytics.ear`:

Locate `analytics.ear` in the directory `OracleBI_HOME/web`.

2. Name the new Plug-in "analyticsSOAP".

Make the same modifications to the `web.xml` file for this `analyticsSOAP` servlet that were made to the `web.xml` file for the default "analytics" servlet. For more information on updating the `web.xml` file, see the topic "Configuring BI Presentation Services Plug-in" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

3. Make the following modification to the file `mod_osso.conf` to open `analyticsSOAP` to requests from the BI Office Server:

(The `mod_osso.conf` is located in the directory `Oracle_HOME/Apache/Apache/conf`.)

```
<Location /analyticsSOAP>
  Require valid-user
  AuthType Basic
  Allow from <IP address of BI Office server>
  Satisfy Any
</Location>
```

where

<IP address of BI Office Server> is the IP address of the BI Office Server, for example: 10.155.224.7

4. Locate the BI Office Server configuration file in the following directory:
`OAS_HOME\j2ee\home\applications\biooffice\biooffice\WEB-INF`
5. Update the following properties:
 - "SawBaseURL" to point to the new `analyticsSOAP` deployed in the previous steps
 - "SawWebURLforSSO" to point to the URL used for public and external users to access the Oracle BI Answers page.

Note: When specifying the URL for `SawWebURLforSSO`, you can use the domain names in the URL. However, note the following special cases:

- If the Oracle BI Server, the BI Office Server, and the client computer run in different domains, then you must specify the complete domain name. For example, you would specify:
`http://myserver.mycompany.com:443/analytics/saw.dll`
 not
`http://myserver:443/analytics/saw.dll`
 - If the BI Server or the BI Office Server is running on a computer that has two or more IP addresses configured, then you must specify the correct IP address in the URL for the location of the Oracle BI Server.
-

- "SawSSOImpersonator" with the Impersonator user that you set up for the Oracle BI Server.
- "SawSSOImpersonatorPassword" with the password for the Impersonator user.
- "SawUseSSO" to indicate that you have enabled SSO for Oracle BI for Microsoft Office and Presentation Services. Set the value to "1".

Example:

```
<property
name="SawBaseURL">http://localhost:port/analyticsSOAP/saw.dll</property>
<!-- Does SAW use SSO (Single Sign-On): yes = 1; no = 0; -->
<property
name="SawWebURLforSSO">http://localhost:port/analytics/saw.dll</property>
<property name="SawSSOImpersonator">Impersonator</property>

<property name="SawSSOImpersonatorPassword">password</property>
<property name="SawUseSSO" type="int">1</property>
```

For more information on the `biooffice.xml` file, see [Section 31.5, "Setting Properties in the biooffice.xml Configuration File"](#).

31.8.3 Alternative Configuration Method

As an alternative to deploying a second Presentation Services Plug-in, you can perform the following configuration steps:

1. Modify the `web.xml` file (located at `OracleBI\oc4j_bij2ee\home\applications\analytics\analytics\WEB-INF\web.xml`) by entering a duplicate of the servlet `SAWBridge` and the servlet mappings. Give a different servlet name and URL pattern to your duplicate entry.
2. Restart the Web server. Now you have two different servlets with two different URL patterns.
3. Follow the instructions in the [Section 31.8.2](#), starting with Step 3 to unprotect the second servlet and configure the `biooffice.xml` file.

Following is a sample `web.xml` file showing the duplicate entry:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE web-app
PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.2//EN"
http://java.sun.com/j2ee/dtds/web-app_2_2.dtd">

<web-app>
  <servlet>
    <servlet-name>SAWBridge</servlet-name>
    <servlet-class>com.siebel.analytics.web.SAWBridge</servlet-class>
    <init-param>
      <param-name>oracle.bi.presentation.sawserver.Host</param-name>
      <param-value>localhost</param-value>
    </init-param>
    <init-param>
      <param-name>oracle.bi.presentation.sawserver.Port</param-name>
      <param-value>9710</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>SAWBridge</servlet-name>
    <url-pattern>/saw.dll</url-pattern>
  </servlet-mapping>

  <servlet>
    <servlet-name>SAWBridge2</servlet-name>
    <servlet-class>com.siebel.analytics.web.SAWBridge</servlet-class>
    <init-param>
      <param-name>oracle.bi.presentation.sawserver.Host</param-name>
      <param-value>localhost</param-value>
    </init-param>
    <init-param>
      <param-name>oracle.bi.presentation.sawserver.Port</param-name>
      <param-value>9710</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>SAWBridge2</servlet-name>
    <url-pattern>/soap</url-pattern>
  </servlet-mapping>

  <mime-mapping>
    <extension>xsd</extension>
    <mime-type>text/xml</mime-type>
  </mime-mapping>
  <welcome-file-list>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
</web-app>
```

31.9 Setting Up an NTLM-Enabled Presentation Server for Releases 10.1.3.4 and Later

Oracle BI for Microsoft Office connections running on supported Windows client operating systems that use Active Directory as the LDAP server for Windows authentication can now log in to Oracle BI EE without requiring the user to enter a login user name and password, if Oracle BI EE is configured to authenticate against the same Active Directory server.

NT LAN Manager (NTLM) Authentication Protocol is the name of a family of security protocols in Windows. Starting with Release 10.1.3.4, Oracle BI for Microsoft Office supports NTLM protocol to achieve the following features:

- Single user repository - When using Microsoft systems, Oracle BI for Microsoft Office can leverage NTLM protocol to authenticate users against the Windows user repository, eliminating the need for a separate user repository for Oracle BI for Microsoft Office and the Oracle BI EE applications.
- Silent Sign-on -When accessing the add-in from an authorized computer, a user is not challenged for credentials. The NTLM protocol authenticates the user and silently logs the authenticated user on to the system.

For details on NTLM, refer to the Microsoft Web sites at www.microsoft.com and msdn.microsoft.com.

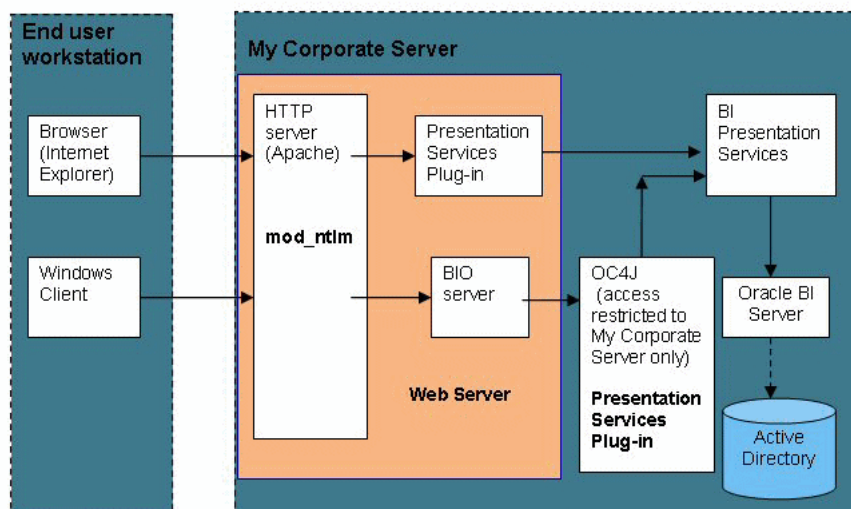
31.9.1 Example of Deploying and Configuring with NTLM Authentication

This section provides an example of deploying and configuring Oracle BI for Microsoft Office with NTLM authentication. This example uses the following components:

- Windows Operating System
- Apache HTTP Server
- OC4J Web server

Deployment and configuration steps might differ if you choose different components for your system.

The following diagram displays a sample deployment topology:



Features of this topology:

- Every external HTTP request goes through the HTTP server with NTLM enabled. These requests are authenticated by the NTLM module.
- The BI Office Server and Presentation Services Plug-in modules, and Presentation Services and Oracle BI Server, are deployed behind the HTTP server. These components therefore cannot be accessed without passing HTTP server authentication.
- In this example a second Presentation Services Plug-in is deployed on a dedicated instance of an OC4J Web server. Access to the Web server is restricted to protected

IP addresses, including the IP address of the computer where the BI Office Server runs. This configuration enables the BI Office Server to send Web service requests directly to the Presentation Services Plug-in without going through Apache NTLM authentication.

- Optionally, you can use the BI Administration Tool to configure the Oracle Business Intelligence user repository (rpd) to use Microsoft Active Directory or another third-party LDAP server to achieve a single user repository. If you choose not to configure a single repository, then the Oracle Business Intelligence user repository must duplicate the Microsoft system user repository to achieve silent sign-on. For example, Windows user AJones with password abc123 must be duplicated in the Oracle Business Intelligence user repository. For information on configuring the user repository, see *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

31.9.2 Prerequisite Tasks

Following are the prerequisite tasks:

31.9.2.1 Configure Oracle BI EE to Enable Single Sign-On

For details see the chapter, "Enabling Oracle Single Sign-on for Oracle Business Intelligence" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

31.9.2.2 Optionally Install and Configure Active Directory or Another Third-Party LDAP Server

To achieve single user repository management, you can use the BI Administration Tool to configure the Oracle Business Intelligence user repository (rpd) to use Microsoft Active Directory or another third-party LDAP server. If you choose not to configure a single repository, then the Oracle Business Intelligence user repository must duplicate the Microsoft system user repository to achieve silent sign-on. For information on configuring the user repository, see *Oracle Business Intelligence Enterprise Edition Deployment Guide*

31.9.2.3 Create a Second BI Presentation Services Servlet to Bypass Authentication

There are two methods for doing this:

- Install a standalone instance of the Web server, deploy the Presentation Services Plug-in to this Web server, and configure as described below.
- Create a duplicate SAWBridge servlet in the web.xml file as described below.

31.9.2.3.1 Method 1: Install a Standalone Instance of the Web Server and Deploy and Configure the Presentation Services Plug-in

1. On the same computer where the BI Presentation Services Plug-in has been deployed, install an instance of OC4J or other supported Web server.
2. Deploy the Presentation Services Plug-in to this new Web server instance using the file analytics.ear:

Locate analytics.ear in the directory OracleBI_HOME/web.

3. After the plug-in has been deployed, make the same modifications to the web.xml file for this servlet that were made to the web.xml file for the default "analytics" servlet. For more information on updating the web.xml file, see the topic

"Configuring BI Presentation Services Plug-in" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*.

4. Apply access restriction to this Web server so that it only accepts requests from the protected IP address where the BI Office server is installed.

31.9.2.3.2 Method 2: Create a Duplicate SAWBridge Servlet in the web.XML File Rather than deploy a second Web server and Presentation Services Plug-in you can create a duplicate servlet directly within the web.xml file.

1. Locate the web.xml file as follows:

```
OracleBI\oc4j_
bi\j2ee\home\applications\analytics\analytics\WEB-INF\web.xml
```

2. Duplicate the entries for the SAWBridge servlet and the servlet mappings. Following is a sample servlet and servlet mapping from the web.xml file:

Note: For more information on updating the web.xml file, see the topic "Configuring BI Presentation Services Plug-in" in *Oracle Business Intelligence Enterprise Edition Deployment Guide*

```
<servlet>

<servlet-name>SAWBridge</servlet-name>
<servlet-class>com.siebel.analytics.web.SAWBridge</servlet-class>
<init-param>
<param-name>oracle.bi.presentation.sawserver.Host</param-name>
<param-value>localhost</param-value>
</init-param>
<init-param>
<param-name>oracle.bi.presentation.sawserver.Port</param-name>
<param-value>9710</param-value>
</init-param>
</servlet>
<servlet-mapping>
<servlet-name>SAWBridge</servlet-name>
<url-pattern>/saw.dll</url-pattern>
</servlet-mapping>
```

3. Give your duplicate servlet a new name and URL pattern. For example:

```
<servlet-name>SAWBridge2</servlet-name>

<url-pattern>/soap</url-pattern>
```

4. Restart the Web server (for example: oc4j or IBM Websphere).

Now you have two Presentation Services plug-in servlets with two different URL patterns. You can protect one with SSO and unprotect the other.

31.9.3 Installing the NTLM Module

The NTLM authentication module is an open source application. There are several implementations available that can be found on the Web. This document describes how to install NTLM as an Apache module on a Windows operating system. For different implementations, see the documentation that accompanies the specific system components.

The NTLM module described in this document is downloadable from this location:
<http://sourceforge.net/projects/mod-auth-sspi>

Perform the following to download and install the NTLM module:

1. Download the NTLM module.

Note: For UNIX operating systems, download mod_ntlm from:
http://www.jamiekerwick.co.uk/?page_id=5. The installation guide is available from this link: <http://modntlm.sourceforge.net/>

2. Install the NTLM module to the application server's Apache HTTP server, as described in the installation documentation. Following is a brief overview of the install steps:

- a. Copy the mod_auth_sspi.so module to the Apache modules directory.
- b. Update the Apache configuration file, httpd.conf, to load the module. Then add location directives to protect the biooffice and the analytics applications.

Following is an example of the updates to the httpd.conf file:

```
...
# Load ntlm module
LoadModule sspi_auth_module modules/mod_auth_sspi.so
...
#Add location protection like this:

# Configuration for mod_auth_sspi
<IfModule mod_auth_sspi.c>
  <Location /biooffice/>
    AuthName "A Protected Place"
    AuthType SSPI
    SSPIAuth On
    SSPIAuthoritative On
    SSPIOfferBasic On
    # SSPIBasicPreferred
    # SSPIUsernameCase lower
    require valid-user
  </Location>

  <Location /analytics/>
    AuthName "A Protected Place"
    AuthType SSPI
    SSPIAuth On
    SSPIAuthoritative On
    SSPIOfferBasic On
    # SSPIBasicPreferred
    # SSPIUsernameCase lower
    require valid-user
  </Location>
</IfModule>
# End of mod_auth_sspi
```

31.9.4 Configuring Oracle BI for Microsoft Office to Enable NTML

Perform the following steps to enable NTLM for Oracle BI for Microsoft Office.

1. Locate the BI Office Server configuration file (biooffice.xml) in the following directory:

OAS_HOME\j2ee\home\applications\biooffice\biooffice\WEB-INF

2. Update the following properties:

- "SawBaseURL" to point to the new Presentation Services Plug-in that you installed on the dedicated Web server. Web service requests from Oracle BI for Microsoft Office to the Oracle BI Server are routed through the Presentation Services Plug-in that is specified in this property. These requests bypass NTLM authentication.
- "SawWebURLforSSO" to point to the URL used for public and external users to access the Oracle BI Answers page.

Note: When specifying the URL for SawWebURLforSSO, you can use the domain names in the URL. However, note the following special cases:

- If the Oracle BI Server, the BI Office Server, and the client computer run in different domains, then you must specify the complete domain name. For example, specify:
https://myserver.mycompany.com:443/analytics/saw.dll
not
https://myserver:443/analytics/saw.dll
 - If the BI Server or the BI Office Server runs on a computer that has two or more IP addresses configured, then you must specify the correct IP address in the URL for the location of the Oracle BI Server.
-

- "SawSSOImpersonator" with the Impersonator user you set up for the Oracle BI Server.
- "SawSSOImpersonatorPassword" with the password for the Impersonator user.
- "SawUseSSO" to indicate that you have enabled SSO for Oracle BI for Microsoft Office and Presentation Services. Set the value to "1".

Example:

```
<property
name="SawBaseURL">http://152.68.81.15:443/analytics/saw.dll</property>
<!-- Does SAW use SSO (Single Sign-On): yes = 1; no = 0; -->
<property
name="SawWebURLforSSO">http://152.68.196.231:443/analytics/saw.dll</property>
<property name="SawSSOImpersonator">Impersonator</property>

<property name="SawSSOImpersonatorPassword">password</property>
<property name="SawUseSSO" type="int">1</property>
```

For more information on the biooffice.xml file, see [Section 31.5, "Setting Properties in the biooffice.xml Configuration File"](#).

31.9.5 Testing the NTLM Deployment

After you have completed the setup steps, to test the deployment, access Oracle BI for Microsoft Office from a computer within the authenticated Microsoft system domain. You are silently signed in without being challenged for credentials.

Select the **Edit View** link for a request. You are silently signed in to Presentation Services and the Answers request Edit page is displayed.

Microsoft Office Support in Release 10.1.3.3

This new feature enables you to analyze and deploy business intelligence reports more easily in Microsoft Office.

This release of Oracle BI EE features an improved version of Oracle Business Intelligence Add-in for Microsoft Office, with a new Excel and PowerPoint Add-in. These add-ins allow you to insert Oracle BI EE views as native Office tables and BI chart views as native Office charts, as static but refreshable images, or even as high-quality Flash objects, leveraging Office functionality while referencing the Oracle BI semantic layer, thus ensuring that a single version of the truth is available to users across an enterprise.

Oracle BI for Microsoft Office provides the following functionality:

- Oracle BI Add-in for Microsoft Excel.
- Oracle BI Add-in for Microsoft PowerPoint.
- When installed on Excel 2007, the Oracle BI Add-in's functionality is made available through a native Office ribbon interface. When installed on Excel 2003, the Oracle BI Add-in's functionality is made available through a menu and a toolbar. In both cases, an Office pane is available for browsing the BI Presentation Catalog and for selecting views to insert inside Office documents.
- Oracle BI Office Server. This is a server-side J2EE application that communicates between the add-in and Oracle BI Presentation Services using Web services.
- Ability to secure Oracle BI EE data in BI views inserted inside Excel spreadsheets or PowerPoint presentations. Metadata that describes the BI view is preserved, but all data is removed by the BI Add-in. You can obtain the latest data available by refreshing BI views using the BI Add-in.
- You can copy views from Answers and Interactive Dashboards pages and then paste them into Excel and PowerPoint documents using the "Paste" feature of the Oracle BI Add-in. Copied views are pasted as Office documents: Excel or PowerPoint tables and charts. This functionality also extends to copying compound views from Answers and Interactive Dashboards. Compound views copied from Answers or Interactive Dashboards are pasted into Office documents as native Office tables and charts. Only BI views supported by the Oracle BI Add-in are pasted into Office documents. Gauge and funnel views are pasted as refreshable images when using this feature in Release 10.1.3.4.
- You can use this "Copy" feature to copy the state of BI views in Interactive Dashboards and paste this state into Office documents. You can drill on a chart from year to quarter, then drill from region to the district level, using the "Copy" link, if available for that view, to copy this state and then paste it into an Office document using the add-in. When you refresh the BI views inside the Office

document, the data is refreshed and displayed based on this state of the request. Prompt definitions are preserved when using this feature in Release 10.1.3.4 and can be changed subsequently in Excel or PowerPoint documents.

For more information on how the "Copy" feature works from Answers and Dashboards, see the previous item that describes the "Paste" feature.

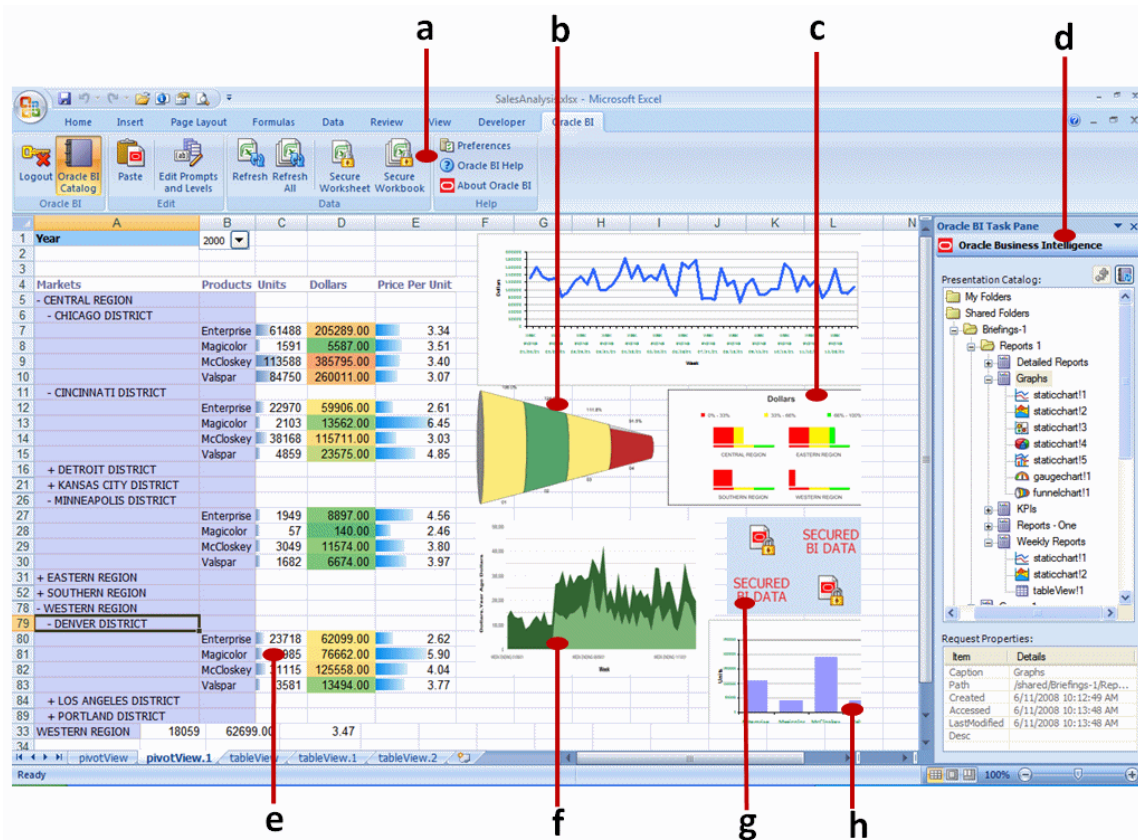
- In versions of Oracle BI for Microsoft Office before 10.1.3.4, a "Flat Data View" is provided for every request. This view provides the entire data set for the original request in a simple tabular format. This view is more suited when you want to obtain only the data and then use Excel's functionality to perform further analysis on the data.
- The table view displays results in a tabular format. You can add totals, customize headings, and change the formula or aggregation rule for a column. (In releases before 10.1.3.4, column drilling is also supported in Excel tables. If users have the add-in installed, then they can make use of this drilling functionality even in offline mode, without being connected to Presentation Services.)
- The pivot table view, available in the add-in for Excel, presents data from Oracle BI EE requests with page items and a section-based layout if these had been defined in the BI pivot table view for the request in Answers.
- Support for the Oracle BI Security model, including support for both encrypted (SSL) and Single-Sign On modes. The same login that you use for Answers or Interactive Dashboards is available for the add-in. For Release 10.1.3.4 and later, if your environment is NTLM-enabled, then you can use the add-in's support for NTLM-based authentication to enable silent and single sign-on.
- The BI Presentation Catalog browser is presented as a pane in Excel and PowerPoint to browse requests, both user-created and shared. You can also display the individual views that are available for each request.
- Ability to insert table, pivot table, and chart views into Excel documents as Excel tables and Excel charts (subject to the limitations of Excel).
- Support for prompts that are defined in the Oracle BI request.
- If a view selected for insertion has multiple levels from one or more dimensions, then you can choose to limit the data for each dimension by unchecking these levels from a "level selection" dialog.
- Ability to insert table and chart views into PowerPoint presentations as PowerPoint tables and charts (subject to the limitations of PowerPoint).
- Ability to insert chart views as images inside Excel spreadsheets and PowerPoint presentations that can be refreshed using the add-in.
- Ability to insert chart views as high-quality Flash objects inside PowerPoint presentations that can then be refreshed using the add-in.
- For Release 10.1.3.4, the ability to insert Oracle BI EE gauge views as an image or as a Flash object. Flash support for gauges is available only in PowerPoint.
- For Release 10.1.3.4, the ability to insert Oracle BI EE funnel views as an image in Excel and PowerPoint.
- You can define Excel-specific conditional formats to data from BI views. These conditional formats are preserved during data refreshes.
- You can change the formats of data items that are displayed in BI views in Excel and PowerPoint documents. To change the format of an item in an Excel sheet, you must modify the style for that cell. These formatting changes are then preserved

even during refreshes. Furthermore, these formatting changes are applied to all cells with the same style.

- For BI chart views inserted as native Excel or PowerPoint charts, you can change the chart type and apply other formatting changes using Excel and PowerPoint charting capabilities. These changes are preserved during refreshes.
- Ability to customize the default appearance and placement of BI tables and pivot table views that are inserted in Excel sheets using an Excel template installed during the client add-in installation.

Figure 32–1 shows the Oracle Business Intelligence Add-in for Microsoft Excel.

Figure 32–1 Oracle BI Add-In for Excel

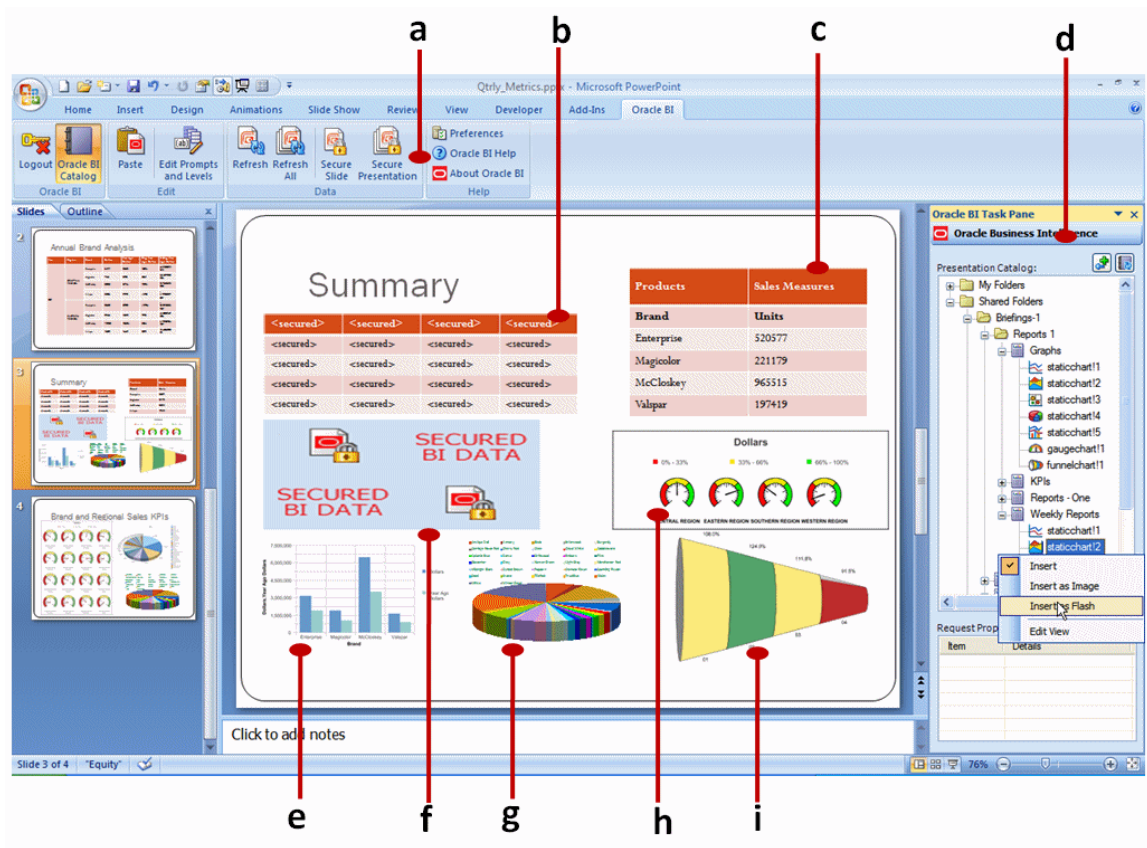


The following list describes the callout text in Figure 32–1:

- a - Oracle BI for Microsoft Office Add-in ribbon in Excel 2007
- b - BI funnel view inserted as refreshable image
- c - BI gauge view inserted as refreshable Flash object
- d - BI Catalog with list of requests and views available for insertion
- e - BI pivot table view inserted as sectioned, refreshable table with page-item list
- f - BI chart view inserted as refreshable image
- g - BI chart view inserted as refreshable image and then secured
- h - BI chart view inserted as refreshable Office chart

Figure 32–2 shows the Oracle Business Intelligence Add-in for Microsoft PowerPoint.

Figure 32-2 Oracle BI Add-In for PowerPoint



The following list describes the callout text in [Figure 32-2](#):

- a - Oracle BI for Microsoft Office ribbon in PowerPoint 2007
- b - BI table view inserted as PowerPoint table and then secured
- c - BI table view inserted as PowerPoint table
- d - BI Catalog with list of requests and views available for insertion
- e - BI chart view inserted as refreshable image
- f - BI chart view inserted as refreshable image and then secured
- g - BI chart view inserted as refreshable Office chart
- h - BI gauge view inserted as refreshable Flash object
- i - BI funnel view inserted as refreshable image

New Features in Oracle BI for Microsoft Office Release 10.1.3.3.1

Oracle BI EE Release 10.1.3.3.1 introduces the following new features for Oracle BI for Microsoft Office:

- [Section 33.1, "Configuring the Connection URL"](#)
- [Section 33.2, "Language Support"](#)

33.1 Configuring the Connection URL

When you deploy the BI Office Server application file to OC4J or to a supported J2EE container, you can provide any name for the application. The default is "biooffice", but you can specify any other name such as "myoffice", "financeoffice", "logisticsoffice". In 10.1.3.3.0, the URL that the Oracle BI Add-in for Microsoft Office connects to always appended the default application name "biooffice" to the server (for example: `http://mybiserver.mycompany.com:9704/biooffice/...`), and there was no way to change the application name appended to the end of the URL.

In 10.1.3.3.1 you can now specify an "App name" for the BI Office Server to configure the URL that the add-in connects to. Specify the App Name in the Connection Details dialog available from the Preferences menu selection. The value for the App Name field is used to construct the add-in's connection URL. For example:
`http://mybiserver.mycompany.com:9704/financeoffice/...` or
`http://mybiserver.mycompany.com:9704/logisticsoffice/...`

The following image shows an example of the Connections Details dialog:



33.2 Language Support

Release 10.1.3.3.1 of Oracle BI for Microsoft Office adds the capability to support all the languages that Oracle BI EE supports. This support is reflected in the login dialog also

where you can select from any of the approximately 25 languages. If Oracle BI EE metadata is available in multiple languages then Oracle BI for Microsoft Office also displays this metadata in the language selected. For example, the language that you select to display Oracle BI EE metadata such as dimensions (such Country or Brand), items, and measures (such as Sales or Profits) is used when you connect to this Oracle BI EE instance from the add-in.

New Features in Oracle BI for Microsoft Office Release 10.1.3.3.3

Oracle BI EE Release 10.1.3.3.3 introduces no features for Oracle BI for Microsoft Office. However, additional configuration instructions for Single Sign-On have been included in this document. See [Chapter 31, "Installing Oracle BI for Microsoft Office in Release 10.1.3.3"](#) for information.

New Features in Oracle BI for Microsoft Office Release 10.1.3.4

Oracle BI EE Release 10.1.3.4 introduces the following new features for Oracle BI for Microsoft Office:

- [Section 35.1, "Server Independence"](#)
- [Section 35.2, "Changes to the Add-In"](#)
- [Section 35.3, "Support for Combined Request Views"](#)
- [Section 35.4, "Transparent Compression of Results"](#)
- [Section 35.5, "NTLM Support"](#)

35.1 Server Independence

Views inserted in Excel and PowerPoint documents by the Oracle BI Add-in for Microsoft Office connected to one server can now be refreshed by connecting the add-in against a different server, if the view definitions are the same.

35.2 Changes to the Add-In

The following changes were made for the Oracle BI Add-in for Microsoft Office:

- [Section 35.2.1, "Ability to Enable Logging"](#)
- [Section 35.2.2, "Enhanced BI Catalog Task Pane"](#)
- [Section 35.2.3, "Ability to Preserve Excel Conditional Formatting During Refreshes"](#)
- [Section 35.2.4, "Support for Gauge View"](#)
- [Section 35.2.5, "Support for Funnel View"](#)
- [Section 35.2.6, "Edit Prompts and Levels After Insert"](#)
- [Section 35.2.7, "Enhanced Prompts Selection User Interface"](#)
- [Section 35.2.8, "Support for Grand Totals in Table Views"](#)

For more information about these changes, see the add-in's Help system.

35.2.1 Ability to Enable Logging

You can enable logging in the add-in on the Advanced tab of the Preferences dialog. Log messages are written in an HTML format to files in the add-in's installation folder.

35.2.2 Enhanced BI Catalog Task Pane

Chart views in the BI Catalog are identified by a chart type icon that is specific to that chart type. There are icons for bar, area, horizontal bar, line, line-bar combo, pareto, pie, and scatter chart types. These icons help you identify a chart type before inserting it into an Office document.

35.2.3 Ability to Preserve Excel Conditional Formatting During Refreshes

The add-in can preserve any conditional formatting on data that you define using Excel functionality. This requires the add-in to perform cell-level comparisons to preserve the formatting while updating the cell values with updated data, which is performance-intensive. To improve performance, use an option that specifies whether to preserve Excel conditional formatting during refreshes. Select the **Preserve Conditional Formatting** option in the General tab of the Preferences dialog.

35.2.4 Support for Gauge View

You can add gauge views to PowerPoint and Excel documents. The gauge view is available for insertion from the BI Catalog in the following formats:

- Image: in Excel and PowerPoint
- Flash: in PowerPoint only

If you use the Copy to Clipboard feature in Answers and Dashboards and then paste the view in a BI Office document, then the view is inserted as an image.

35.2.5 Support for Funnel View

You can add funnel views to PowerPoint and Excel documents. The funnel view is available for insertion from the BI Catalog as an image in both Excel and PowerPoint.

35.2.6 Edit Prompts and Levels After Insert

When you insert a view in an Excel or PowerPoint document, you can edit or change the levels or prompts for that view. To do so, select the view and click the **Edit Prompts and Levels** menu item, toolbar button, or ribbon item.

You can edit or change the levels or prompts for the following types of views:

- Views that you inserted using the BI Catalog task pane inside Excel and PowerPoint.
- Views that you copied using the **Copy Request XML to Clipboard** link in Answers and Dashboards and pasted into Excel and PowerPoint documents using the add-in's **Paste** option.

35.2.7 Enhanced Prompts Selection User Interface

Support for prompts has been enhanced to provide the following prompt properties that are available in Answers:

- Multi-selection of values
- Search for prompt values
- Shuttle UI for selecting multiple values
- Cascading prompts where the value of one prompt limits the values available in a subsequent prompt (Year = 2008 restricts Quarters to only Q1)

- Skip prompts
- Enter ranges of values

35.2.8 Support for Grand Totals in Table Views

You can enter grand totals in table views.

35.3 Support for Combined Request Views

Oracle BI for Microsoft Office supports inserting Answers request views to Excel and PowerPoint that have been constructed using Set operations. Set operations combine results from two or more requests into a single request. The request can be constructed using any of the following Set operations: Union, Union All, Intersect, or Minus. For information on using Set operations to build an Answers request, see the "Combining Oracle BI Request Criteria Using Set Operations" section in *Oracle Business Intelligence Answers, Delivers, and Interactive Dashboards User Guide*.

Note that if you added a grand total when you created the combined request in Answers, the grand total is not displayed when the view is inserted to Excel. Subtotals are displayed.

35.4 Transparent Compression of Results

New settings have been added in the BI Office Server configuration file that enable an administrator to specify whether compression of results should take place before they are transmitted to the add-in, and to specify the minimum size of the result set beyond which the results should be compressed. The add-in detects this result set as compressed and decompresses the result, transparently to and without any intervention on the part of the user.

35.5 NTLM Support

Oracle BI for Microsoft Office connections that run on supported Windows client operating systems that use Active Directory as the LDAP server for Windows authentication can now connect to Oracle BI EE without requiring the user to enter a user name and password, if Oracle BI EE is configured to authenticate against the same Active Directory server.

NT LAN Manager (NTLM) Authentication Protocol is the name of a family of security protocols in Windows. In Release 10.1.3.4, Oracle BI for Microsoft Office supports the NTLM protocol to achieve the following features:

- Single user repository: Oracle BI for Microsoft Office leverages the NTLM protocol to authenticate against the Windows user repository. The client must no longer maintain a separate user repository for the Oracle BI EE or Oracle BI for Microsoft Office application.
- Silent Sign-on: When accessing the add-in from an authorized computer, a user is no longer explicitly asked to log in. The NTLM protocol verifies the user's credentials in the background and silently logs in the authenticated user.

To enable NTLM-based or another type of single sign-on, you must make the appropriate changes to the biooffice.xml configuration file. For details on NTLM, refer to the following:

- [Chapter 31, "Installing Oracle BI for Microsoft Office in Release 10.1.3.3"](#)

- msdn.microsoft.com
- www.microsoft.com

New Features in Oracle BI for Microsoft Office Release 10.1.3.4.1

Oracle BI EE Release 10.1.3.4.1 introduces no features for Oracle BI for Microsoft Office.

New Features in Oracle BI for Microsoft Office Release 10.1.3.4.2

Oracle BI EE Release 10.1.3.4.2 introduces the following new features for Oracle BI for Microsoft Office:

- Features for views:
 - The **Re-insert if the view changes on server** option has been added to the General tab of the Preferences dialog.
 - Filter views are displayed as check boxes in the Insert View dialog in Office documents. Filter views are inserted with the request into Office documents and are refreshed from the Oracle BI Server.
 - Title views are displayed as check boxes in the Insert View dialog in Office documents. Title views are inserted with the request into Office documents and are excluded from subsequent refreshes from the BI Server.
 - The option to paste a chart as an image is available in Excel and as an image or Flash format object in PowerPoint documents.
 - The default chart sub-type that you specify in Answers is applied when charts are inserted into Office documents.
 - The Line-Bar Combo chart is displayed in Office documents in the same way as it is in a request. For example, if you see two bars and two lines on a Line-Bar Combo chart in Answers, then you see the same bars and lines in the chart in Excel.
- Features for the Oracle BI Add-in for Excel:
 - The **Always insert Pivot Table fully expanded** option in the General tab of the Preferences dialog is selected by default.
 - In Excel documents, pivot table views support subtotals and table views support subtotals and grand totals.
- Features for the Oracle BI Add-in for PowerPoint:
 - A pivot table view is inserted as a simple crosstab in PowerPoint documents.
 - The initial display formatting for a pivot table is imported from the BI Server.
 - After you insert a table or pivot table into a PowerPoint document, you can apply formatting to it. This custom formatting is preserved if you refresh the view from the BI Server.
 - In PowerPoint documents, pivot table views and table views support subtotals and grand totals.

