



Siebel Reports Guide

Siebel 2018

April 2018

ORACLE®

Copyright © 2005, 2018 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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

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

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

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 unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Documentation Accessibility

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

Access to Oracle Support

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

Contents

Chapter 1: What's New in This Release

Chapter 2: Overview of Siebel Reports

About Siebel Reports	11
Types of Users Who Use Siebel Reports	11
Scenario for Customizing Reports	12
How Users Run Reports	13
Output File Types That Siebel Reports Supports	14
How Siebel CRM Creates Reports	14
About Oracle BI Publisher Desktop	15
Overview for Using This Book	17

Chapter 3: Running Reports

Running Reports in Siebel CRM Clients	19
Setting User Preferences for Siebel Reports	19
Running Reports	20
Monitoring Report Status	22
Viewing Report Output	23
Deleting Reports	24
Scheduling Reports	25
Monitoring, Suspending, Viewing, and Deleting Scheduled Reports	29
Using Report Parameters to Filter Reports	29

Chapter 4: Integrating Oracle BI Publisher with Siebel CRM for Siebel Innovation Pack 2017

About Integrating Oracle BI Publisher with Siebel CRM	31
Requirements for Using Siebel Reports for First-Time Installations	32
Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations	33
Preparing for Integration of Oracle BI Publisher with Siebel CRM	35
Process of Installing Oracle BI Publisher for Integration with Siebel CRM	36
Installing Oracle Business Intelligence Publisher Enterprise	37

Installing Oracle Business Intelligence Publisher Add-in for Microsoft Word	38
Copying JAR Files to Oracle BI Publisher Server	39
Enabling External File References on Oracle BI Publisher Server	40
Starting and Stopping Oracle WebLogic Server and Oracle BI Publisher	40
Upgrading to the Latest Oracle BI Publisher Patch	41
Configuring an Outbound Web Service for Siebel Reports	41
Enabling Siebel Server Component Groups for Siebel Reports	42
About Security and Authentication for Siebel Reports	43
About Oracle Business Intelligence Enterprise Edition Coexistence with Siebel Reports	44
About Configuring Security and Authentication for Siebel Reports	45
Configuring Security and Authentication for Siebel Reports	45
Setting Up and Enabling the Siebel Security Model for Siebel Reports	46
Setting Responsibilities in the Siebel Application	47
Configuring Security and Authentication for Siebel Reports Using LDAP	48
Configuring Siebel Reports in a Web Single Sign-On Environment	49
Adding an Explicit Reference to a Container URL for Oracle BI Publisher Server	54
Implementing EAI File Handling Changes for Siebel Reports Generation	55
Uploading Preconfigured Report Layout Templates to Oracle BI Publisher Server	56
Process of Enabling and Configuring Report Scheduling	57
Configuring the Data Service WSDL for Oracle BI Publisher Server	57
Verifying Symbolic URL Arguments That Embed the Oracle BI Publisher Scheduling Views in the Siebel Application	58
About Upgrading Siebel Reports for Siebel Innovation Pack 2017	59

Chapter 5: Administering Siebel Reports

Doing Typical Administrative Tasks	61
Disabling the Reports Button	61
Making Sure Siebel CRM Displays The Report Status	62
Making Sure Fonts Are Available for Reports	62
Managing Report Notifications	63
Viewing Report Usage Statistics	63
Configuring Siebel CRM to Automatically Delete Reports	63
Optimizing the Performance of Siebel Reports	65
Setting the Waiting Period	65
Setting the Polling Interval	66
Setting Concurrency Parameters	66

Optimizing Reports That Include Thousands of Records	67
Administering Multilingual Reports	71
How Siebel CRM Runs Multilingual Reports	71
Creating Multilingual Reports	73
Administering Signatures in Reports	75
Creating and Modifying an Integration Object for Signatures in Reports	75
Customizing a Report Template for Signatures	76

Chapter 6: Siebel Reports Architecture

About the Siebel Reports Architecture	77
Overview of How Siebel CRM Runs Reports in Clients	78
How Siebel CRM Runs Reports in Connected Clients	79
How Siebel CRM Runs Reports in Disconnected Clients	81
How Siebel Reports Uses Integration Objects	82
Computer Languages That Siebel Reports Uses	83
About the Report Business Service	83
Directory Structure That Siebel Reports Uses	86
How Siebel CRM Controls Access to Reports	87

Chapter 7: Customizing Siebel CRM Reports

Process of Creating Custom Reports	90
Creating Report Templates	91
Registering Report Templates	93
Associating Report Templates with Views	97
Specifying the Siebel CRM Data That Report Templates Use	97
Modifying Integration Objects	100
Creating XML Files from Integration Objects	103
Reducing the Amount of Data That Integration Objects Transfer	104
Defining Optional Fields When Registering Report Templates	105
Allowing Users to Choose The Records That a Report Contains	106
Sharing Report Templates	107
Customizing Reports That Use Parameters	107
About Report Parameters	108
Adding Report Parameters to Report Templates	108
Report Parameter Types You Can Specify	111
Adding Scrollbars to the Parameters Section of the Run Report Pane	112
Customizing Other Options for Siebel Reports	114
“Modifying the Sequence of Report Names in the Run Report Pane” on page 114	114
Modifying the Sequence of Report Names in the Run Report Pane	114

Chapter 8: Customizing Master-Detail Reports

- About Master-Detail Reports 115
- Viewing the Report Template of a Master-Detail Report 116
- Creating a Master-Detail Report Template in Microsoft Word 120

Chapter 9: Deploying Reports

- Process of Deploying Reports 123
 - Preparing the Deployment Environment 123
 - Packaging the Report Files in the Development Environment 124
 - Deploying the Report Files to the Production Environment 125
 - Packaging Report Database Records in the Development Environment 126
 - Deploying Report Database Records to the Production Environment 127
- Deploying Integration Objects to the Production Environment 128

Chapter 10: Troubleshooting Siebel Reports

- Enabling and Disabling Logging for Siebel Reports 129
 - Enabling Logging for Siebel Reports in the Siebel Application 129
 - Enabling Logging and Debugging on the Oracle BI Publisher Server 131
 - Enabling and Disabling Debugging in Disconnected Clients 131
- Fixing Class Not Found Errors 132
 - Fixing Class Not Found Errors When Previewing Reports in Microsoft Word 133
- Troubleshooting Error Messages for Siebel Reports 134

Index

1

What's New in This Release

What's New in Siebel Reports Guide, Siebel 2018

No new features have been added to this guide for this release. This guide has been updated to reflect only product name changes.

NOTE: Siebel 2018 is a continuation of the Siebel 8.1/8.2 release.

What's New in Siebel Reports Guide, Siebel Innovation Pack 2017, Rev.A

Table 1 lists the changes in this revision of the documentation to support this release of the software.

Table 1. Updates to the Siebel Reports guide, Siebel Innovation Pack 2017, Rev.A

Topic	Description
Configuring Siebel CRM for Integration with Oracle BI Publisher with Web Single Sign-On	The ASDI security adapter references and the related parameters have been removed.
Enabling Reports Scheduling with Web Single Sign-On	

Additional Changes

The guide is updated to correct or remove obsolete product and component terms.

What's New in Siebel Reports Guide, Siebel Innovation Pack 2017

Table 2 lists the changes in this revision of the documentation to support this release of the software.

Table 2. New Product Features in Siebel Reports Guide, Siebel Innovation Pack 2017

Topic	Description
"Configuring Siebel Reports in a Web Single Sign-On Environment" on page 49	Modified topic. It includes information about configuring Siebel CRM and Oracle Business Intelligence Publisher (Oracle BI Publisher) in a Web Single Sign-On environment.
"Administering Signatures in Reports" on page 75	New topic. It includes information about administering reports that contain signatures.

Table 2. New Product Features in Siebel Reports Guide, Siebel Innovation Pack 2017

Topic	Description
Preloading the Siebel Repository File	Removed topic. This procedure is no longer required in Innovation Pack 2017.
“Deploying Integration Objects to the Production Environment” on page 128	Modified topic. A reference to <i>Siebel Database Upgrade Guide</i> for more information about Siebel Migration and the Migration Application Incremental Runtime Repository Data Service has replaced this procedure.

What's New in Siebel Reports Guide, Siebel Innovation Pack 2016, Rev.A

Table 3 lists the changes in this revision of the documentation to support this release of the software.

Table 3. New Product Features in Siebel Reports Guide, Siebel Innovation Pack 2016, Rev. A

Topic	Description
“Where to Find More Information About Siebel Reports” on page 18	Modified topic. A reference to My Oracle Support for more information on system requirements and supported platform certifications for previous product releases has been added.
“Running Reports” on page 20	Modified topic. Information on how to set search and sort order on an integration object, and information on how to sort report columns on date fields using the canonical function has been added.
“Integrating Oracle BI Publisher with Siebel CRM for Siebel Innovation Pack 2017” on page 31	New chapter. Information on how to integrate Oracle Business Intelligence Publisher (Oracle BI Publisher) 11g with Siebel CRM for report generation for Siebel Innovation Pack 2016 has been added. This information was originally available in 1501378.1 (Article ID) on My Oracle Support.
“Making Sure Fonts Are Available for Reports” on page 62	Modified topic. Information on example file directories for fonts in different UNIX operating systems has been provided.
“Managing Report Notifications” on page 63	Modified topic. Information on disabling report notifications has been added.
“Modifying the HTTP Sleep Time” on page 68	Modified topic. Information on default and maximum values for the ReportHttpSleep parameter has been added.
“Preloading the Siebel Repository File” on page 63	New topic. It describes how to improve performance by preloading the Siebel Repository File.
“Creating Multilingual Reports” on page 73	Modified topic. Information on supporting languages other than English in multilingual reports has been added.
“Creating Report Templates” on page 91	Modified topic. Information on saving report templates in RTF format has been added.

Table 3. New Product Features in Siebel Reports Guide, Siebel Innovation Pack 2016, Rev. A

Topic	Description
“Registering Report Templates” on page 93	Modified topic. Information on uploading files when registering templates has been added.
“How Siebel CRM Uploads Report Templates” on page 95	Modified topic. Information on how to locate and on how to upload RTF and XLIFF files when uploading report templates has been added.
“Associating Report Templates with Views” on page 97	Modified topic. Information on how view association is restricted to the primary business object has been added.
“Specifying the Siebel CRM Data That Report Templates Use” on page 97	Modified topic. Information on how to use multiple repositories when specifying integration objects in report templates has been added.
“Reducing the Amount of Data That Integration Objects Transfer” on page 104	Modified topic. Information on optimizing performance during integration object transfer has been added.
“Setting the Log Level for the XMLP Report Server Component” on page 129	Modified topic. Information on how logging levels in the XMLP Report server affect performance has been added.

Additional Changes

This book also contains the following changes:

- Content in this guide was removed and updated to reflect the following:
 - High interactivity (HI) is no longer available in Siebel Reports.
 - WebLogic replaces Oracle Application Server Containers for Java Platform (OC4J), Enterprise Edition.

Book titles have been updated to reflect current versions of technical publications.

2

Overview of Siebel Reports

This chapter describes an overview of Oracle's Siebel Reports. It includes the following topics:

- [About Siebel Reports on page 11](#)
- [Overview for Using This Book on page 17](#)

About Siebel Reports

Oracle® Business Intelligence Publisher (Oracle BI Publisher) is the reporting module for Siebel CRM. You work with Siebel Reports in the following locations:

- In the Siebel client to run, schedule, and view reports. For more information, see [Chapter 3, "Running Reports"](#).
In Microsoft Word to create and customize report templates. For more information, see ["About Oracle BI Publisher Desktop" on page 15](#).
- In Oracle BI Publisher to administer report schedules, report users, and other Oracle BI Publisher administration tasks. This guide includes information about how to do these tasks.

You can run a report in the following Siebel clients:

- Siebel Web Client
- Siebel Mobile Web Client
- Siebel Developer Web Client

Siebel CRM comes with predefined reports and report templates. You can use these reports, modify them, or create your own custom reports. For a complete list of predefined reports and information on how to download them, see 876284.1 (Article ID) on My Oracle Support.

Types of Users Who Use Siebel Reports

The following types of users can use Siebel Reports:

- **Report user.** Runs, views, monitors, schedules, and deletes reports. For example, the report user in a sales organization might include a sales representative, sales manager, or sales executive. If this user uses a computer that includes the Oracle Business Intelligence Publisher Add-in installed in Microsoft Word, then this user can modify an existing or create a custom report template. For more information, see ["About Oracle BI Publisher Desktop" on page 15](#).

- **Business user.** Does administrative tasks to determine how to get data from various sources. For example, a business user in a sales organization might include a sales manager, business analyst, and so on. An administrative task might include uploading a report template to the Siebel Server, and then registering this template in the client, associating a report with a view, and so on.

A sales manager might be a report user or a business user depending on the Siebel responsibility that Siebel CRM assigns to this user. Your business requirements determine the views that each user can access to develop or administer a report. For information about how Siebel CRM determines if a user can view a report, see [“How Siebel CRM Controls Access to Reports” on page 87.](#)

- **Report administrator.** Installs and configures Siebel Reports, deletes reports, and manages how Siebel CRM translates reports from one language to another language.

Scenario for Customizing Reports

This topic describes a scenario of how you might customize reports in Siebel CRM. You might use reports differently, depending on your business model. It describes how a sales manager can work with a business user to customize an opportunity report. For details about customizing reports, see [Chapter 7, “Customizing Siebel CRM Reports.”](#)

Sales Manager Runs Predefined Reports

A team includes five sales representatives that reside in the western region of a company. Every day, the manager of this team runs the following reports to get the most up-to-date information about opportunities, accounts, and forecasted revenues for each sales representative:

- Opportunity by Sales Rep
- Account List
- Forecast Analysis Details

Sales Manager Modifies a Predefined Report Template

The reports that the sales manager runs provide details about the opportunities that currently reside in the pipeline for each sales representative, and it allows the manager to monitor the progress of each representative through the sales cycle. Towards the end of the quarter, the manager must determine the amount of potential revenue the team might close for the quarter. To do this, the manager requires a report that details the revenue probability and sales stage for the team opportunities. No such report currently exists, so the manager uses Oracle BI Publisher Desktop to modify a report template to indicate the desired report layout. The manager then sends this template to a business user to create the new report. For more information, see [“About Oracle BI Publisher Desktop” on page 15.](#)

Sales Manager Runs the Modified Report

The business user finalizes the custom report, and then registers it with the Opportunity view so that the sales manager can choose it in the Run Report pane. The manager runs the report, and then Siebel CRM displays it in a browser window. The manager can save this report. If the report requires time to finish running, then Siebel CRM displays a message that instructs the manager to access the report from the My Reports view after the report finishes running. The manager can also use this view to monitor the status of the report while it runs. For more information about the My Reports view, see [Chapter 3, “Running Reports.”](#)

How Users Run Reports

To run a report, the user typically navigates to a view, runs a query, clicks the Report button in the application toolbar, and then chooses a report to run. The user can run the report immediately or schedule it to run at a later time. The user can monitor the report while it runs, view prior reports, or delete prior reports. For more information about running a report, see [Chapter 3, “Running Reports.”](#)

Table 4. Differences With Running Reports in Siebel Open UI Compared to High Interactivity

High Interactivity	Siebel Open UI
You use the Reports menu.	You use the Run Report pane.
The Reports drop-down list in the Reports menu displays reports that you can run for a view.	The Report Name field in the Run Report pane displays reports that you can run for a view.

Run Report Pane in Siebel Open UI

[Figure 1](#) includes the Run Report pane that Siebel Open UI displays in the client. It anchors this pane in a view (left). It remains open while you navigate through the client.

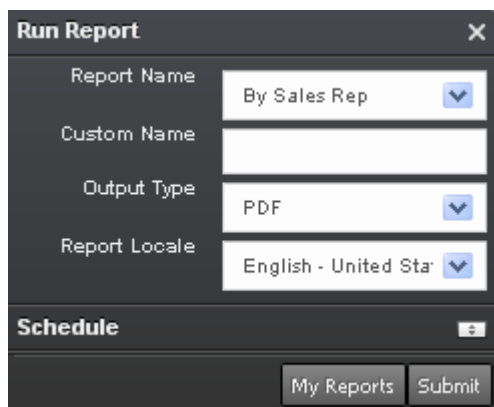


Figure 1. Run Report Pane in the Siebel Open UI Client

How Siebel CRM Displays Reports According to the View



Figure 2. Reports Menu in the High-Interactivity Client

The reports that Siebel CRM displays in the Report Name list of the Run Report pane depends on the view that the client currently displays. For example, if the client currently displays:

- The Accounts List, then the Report Name list displays reports that you can run for accounts.
- The Opportunities List, then the Report Name list displays reports that you can run for opportunities.

The business user can specify these reports that Siebel CRM displays.

Output File Types That Siebel Reports Supports

The user can save a report as of the following file types:

- PDF (Portable Document Format)
- HTML (Hypertext Markup Language)
- RTF (Rich Text Format)
- XLS (Microsoft Excel format)
- PPT (Microsoft PowerPoint format)
- MHTML (MIME HTML)

The PPT and MHTML report output file types are available only with a connected client. For more information, see [“About the Siebel Reports Architecture” on page 77](#).

The business user determines the report output file types that are available. For more information, see [“Registering Report Templates” on page 93](#).

How Siebel CRM Creates Reports

Oracle BI Publisher is the reporting module that Siebel CRM uses to run and manage reports. Siebel CRM installs WebLogic and the Oracle BI Publisher Server when you install Oracle BI Publisher. The following work occurs when a user runs a report from a Siebel application:

- 1 Siebel CRM sends data from the Siebel Database to the Oracle BI Publisher Server.
- 2 The Oracle BI Publisher Server merges the data with the report template from the Oracle BI Publisher repository, and then sends this report to the Siebel client.

For more information about:

- **Oracle BI Publisher.** See the documentation available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).
- **WebLogic.** See *Fusion Middleware Installation Guide for Oracle Business Intelligence* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

About Oracle BI Publisher Desktop

The *Oracle BI Publisher Desktop* is a Microsoft Word plug-in that you can use to create a report template. A *report template* is a type of template that includes layout information for a report, such as the fields that the report contains and the placement of these fields in the report. Siebel CRM saves a report template as an RTF (Rich Text Format) file. Oracle BI Publisher Desktop includes layout, query, and language functionality. It separates logic from the layout, and the layout does not depend on a specific language. It provides flexibility during deployment and helps to reduce maintenance costs. It also includes documentation, demonstrations, and examples.

Figure 3 includes the Oracle BI Publisher toolbar and Oracle BI Publisher menu. Microsoft Word displays this toolbar and menu after you install Oracle BI Publisher Desktop.

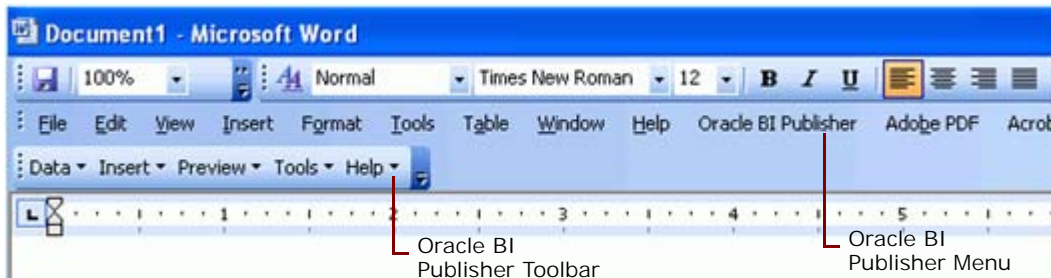


Figure 3. Oracle BI Publisher Toolbar in Microsoft Word

Note the following:

- It is recommended that you use this publisher to create report templates. If you use some other tool to create these templates, then you must manually code the XSL statements and references to the data fields. This book does not describe how to code XSL statements.
- Siebel CRM allows you to use only some of the functionality the Oracle BI Publisher Desktop provides. Oracle only supports the features that this guide describes.
- It is recommended that you use the same version of Oracle BI Publisher Server and Oracle BI Publisher Desktop in your environment.

- The Oracle BI Publisher Desktop is also known as the Oracle BI Publisher Add-in for Microsoft Word.

Advantages of Using Oracle BI Publisher Desktop

Oracle BI Publisher Desktop provides the following advantages:

- You can create the report layout and rules for reports, or you can reuse the report layouts for documents that your organization currently uses.
- Allows a novice user to create a report template in Microsoft Word, and then allows a business user to use this template to finalize the report.
- Makes report customization more efficient because the user can create a report without involving a developer.
- Reduces testing cycles because the user can modify the report without involving a developer.
- Allows the business user to get data, and to get it from multiple sources.
- Allows the business user and report administrator to create and maintain reports.
- Works with other reporting engines that use standards, such as Oracle E-Business Suite, PeopleSoft, JD Edwards, BI EE (Business Intelligence Suite, Enterprise Edition Plus), and so on.

Oracle BI Publisher Desktop Features

Oracle BI Publisher Desktop provides the following features:

- **Template Builder.** Allows you to modify and customize report templates. A user can create a template, and then a business user can use this same template to determine how to get data from the database. These users use the same template and authoring tool, so it is more likely that the business user will accurately interpret the requirements than if these users use different templates and tools. The Template Builder uses Form fields to specify the XSL instructions that Siebel CRM uses to parse XML data. A *Form field* is a feature in Microsoft Word that you use to create a report template in Oracle BI Publisher Desktop. To create these form fields, the Template Builder uses XML code that you create from the Siebel application and the report template. For more information about using the template builder, see [“Process of Creating Custom Reports” on page 90](#).
- **Template Viewer.** Allows you to use sample data to preview the report in multiple output formats.
- **Excel Analyzer.** Allows you to do the following work:
 - Export report query results to an Excel spreadsheet.
 - Log in to Oracle BI Publisher Desktop from Excel to refresh your data, apply new parameters, and apply a template to the report data.
 - Create a template in Excel, upload it to the Oracle BI Publisher repository, and then access and run the report from Excel.

Excel Analyzer is available only in a connected client. For more information, see [“About the Siebel Reports Architecture” on page 77](#).

For more information about using Oracle BI Publisher Desktop, see *Oracle Business Intelligence Publisher Report Designer's Guide* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

Overview for Using This Book

This book uses the following terms, unless noted otherwise:

- The *client* is the Siebel CRM client that a Siebel Business Application uses. This book describes how to use the Siebel Open UI client. Unless noted otherwise, the information that this book describes applies to Siebel Open UI.
- A *user* is a person who uses the client.
- The *server* is the Siebel Server.
- An *administrator* is anyone who uses an administrative screen in the client to configure Siebel Reports. The Administration - Server Configuration screen is an example of an administrative screen.

Computer font indicates a value you enter or text that Siebel CRM displays. For example:

This is computer font

Italic text indicates a variable value. For example, the *n* and the *method_name* in the following format description are variables:

Named Method *n*: *method_name*

The following is an example of this code:

Named Method 2: WriteRecord

A *predefined object* is an object that comes already defined with Siebel CRM. The objects that Siebel Tools displays in the Object List Editor immediately after you install Siebel Tools and the Siebel runtime repository but before you make any customization are predefined objects.

The term *focus* indicates the currently active object in the client. To indicate the object that is in focus, Siebel CRM typically sets the border of this object to a solid blue line.

Depending on the software configuration you purchase, your Siebel Business Application might not include all the features that this book describes.

This book describes the following objects:

- Applets
- Business components
- Business component fields
- Business objects
- Integration components
- Integration objects

■ Views

These objects are part of the Siebel object hierarchy. For more information about this hierarchy, see *Configuring Siebel Business Applications*.

The procedures that this guide describes assumes that you do not use left-hand navigation. However, you can set up left-hand navigation. For more information about left-hand navigation, see *Siebel Fundamentals for Siebel Open UI*.

Where to Find More Information About Siebel Reports

To find more information about:

- Oracle BI Publisher versions that different Siebel versions support, and how to setup Siebel Reports, see 1172844.1 (Article ID) on my Oracle Support.
- Downloading Oracle BI Publisher, see <http://www.oracle.com/technetwork/middleware/bi-publisher/downloads/index.html>.
- General information about Oracle BI Publisher, see <http://www.oracle.com/technetwork/middleware/bi-publisher/overview/index.html>.
- For Siebel CRM product releases 8.1.1.9 and later and for 8.2.2.2 and later, the system requirements and supported platform certifications are available from the Certification tab on My Oracle Support. For information about the Certification application, see article 1492194.1 (Article ID) on My Oracle Support.

3 Running Reports

This chapter describes how to run, view, monitor, and delete reports. It includes the following topics:

- [Running Reports in Siebel CRM Clients on page 19](#)
- [Scheduling Reports on page 25](#)
- [Using Report Parameters to Filter Reports on page 29](#)

Running Reports in Siebel CRM Clients

This topic describes how to run reports in a Siebel CRM client. It includes the following information:

- [Setting User Preferences for Siebel Reports on page 19](#)
- [Running Reports on page 20](#)
- [Monitoring Report Status on page 22](#)
- [Viewing Report Output on page 23](#)
- [Deleting Reports on page 24](#)

Setting User Preferences for Siebel Reports

This topic describes how to set user preferences for Siebel Reports. A user preference that you set overrides the configuration that exists in the report templates.

To set user preferences for Siebel Reports

- 1 Log in to the Siebel client.
- 2 Navigate to the User Preferences screen, and then the Report Parameters view.
- 3 Modify the Locale and Language parameters, as required.

For example, if you set the Locale and Language preferences to German (DEU), then Siebel Reports displays the report output in the German language for all reports. For more information, see [“How Siebel CRM Runs Multilingual Reports” on page 71](#).

- 4 Modify the Output Type parameter.

For example, if you set the Output Type to PDF, then Siebel CRM uses PDF as the report output for all reports. The business user determines the default values that Siebel CRM displays for the report output file types. For more information, see [“Output File Types That Siebel Reports Supports” on page 14](#) and [“Types of Users Who Use Siebel Reports” on page 11](#).

Running Reports

The example in this topic describes how a sales representative can periodically run the following reports:

- Account List
- Pipeline Analysis
- Opportunity Marketing Events Summary

To run reports

- 1 (Optional) Override the default report language and locale.

If a report template includes a default locale or default output type, then Siebel CRM uses this default and overrides the user preferences. For information about overriding report user preferences, see [“Setting User Preferences for Siebel Reports” on page 19](#). For information about how Siebel CRM uses multiple languages, see [“How Siebel CRM Runs Multilingual Reports” on page 71](#).

- 2 Log in to Siebel Call Center connected to the Sample database.

- 3 Navigate to the Account screen, and then the Account list.

- 4 Identify the records that Siebel CRM must include in the report.

Do one of the following:

- Query the Account Type field for Residential.

It is strongly recommended that you do a query to filter records before you run a report. This query limits the number of records that Siebel CRM must process. For more information, see [“Caution About Running Reports with a Large Number of Records” on page 22](#).

- To choose multiple records in a list, hold down the CTRL key while you click each record that Siebel CRM must include in the report. For more information, see [“Using the CTRL Key to Choose Multiple Records for a Report” on page 22](#).

- 5 In the application toolbar, click the Reports button.

Siebel Open UI displays the Run Report pane. The business user determines the default values that Siebel CRM displays for each field that the Run Report pane contains. For more information, see [“Types of Users Who Use Siebel Reports” on page 11](#).

- 6 In the Run Report pane, in the Report Name list, choose Account List.

Siebel CRM displays different reports in the Report Name list, depending on the view that the client currently displays. For more information, see [“How Siebel CRM Displays Reports According to the View” on page 14](#). For information about entering a custom name, see [“Entering a Custom Name” on page 22](#).

- 7 (Optional) In the Output Type list, choose the file type that Siebel CRM uses to save the report.

For example, if you choose HTML, then Siebel CRM saves the report as an HTML file.

- 8 (Optional) In the Report Locale field, choose a language.

For example, if you choose English - United States, then Siebel CRM saves the report in the North American dialect of the English language. You can configure a user preference that overrides the default language. For more information, see [“Setting User Preferences for Siebel Reports” on page 19](#) and [“How Siebel CRM Runs Multilingual Reports” on page 71](#).

- 9 In the Run Report pane, complete the fields, as necessary.

You can click the column heading in a list to sort records before you run a report. Siebel CRM then uses this order when it runs the report. If you configure the sort order on the integration object that Siebel CRM uses to run the report, then Siebel CRM uses the sort order that the integration object specifies. For more information, see [“How Siebel Reports Uses Integration Objects” on page 82](#).

- 10 (Optional) To set search and sort order on an integration object, use the following example syntax:

```
'Account'.Search = "([Name] LIKE ""3*"")" 'Account'.Sort = "Name, Location"
```

NOTE: If you sort records before executing a report, then the sort order is preserved when the report is executed. However, if the sort order is configured on the integration object used to generate the report, then the user-defined sort order is overridden.

- 11 (Optional) To sort report columns on date fields, use the canonical function in the report template. For example:

```
<?sort:psfn:getCanonicalDate(ssHlsCreated);'descending'?>
```

- 12 Click Submit.

Siebel CRM runs the report and then displays a Notification Summary dialog box after it finishes running this report. This dialog box includes a list of the reports that you have recently run. You can click it to access it.

For more information, see [“Managing Report Notifications” on page 63](#).

- 13 Click My Reports.

Siebel CRM displays the My Reports view of the BI Publisher Reports screen. This view displays the report details, such as the report name and status. The report might finish running in a few seconds to a few hours, depending on the number of records and complexity of the report. If a report encounters errors when it runs, then Siebel CRM creates a log file that the business user or administrator can use to troubleshoot the problem. For more information, see [Chapter 10, “Troubleshooting Siebel Reports.”](#)

- 14 Click Account List in the Report Name field.

Siebel CRM displays the report in the format that you specify in [Step 7](#). In this example, it displays the report in a PDF file. It stores the report in the Siebel file system.

- 15 (Optional) Use the Print menu to print the report.

- 16 In the application toolbar, click the Reports button to hide the Run Report pane.

To access this report again, you can navigate to the BI Publisher Reports screen, My Reports view, and then click the report in the Report Name field.

Using the CTRL Key to Choose Multiple Records for a Report

A business user must specify that you can use the CTRL key to choose multiple records for a report. For more information, see [“Allowing Users to Choose The Records That a Report Contains” on page 106](#).

If you do one of the following:

- Press the CTRL key to choose multiple records for a report, then Siebel CRM includes information only from these records in the report.
- Query a list in the client, such as the Account list, or if you use a predefined query, and if you drill down on a record from this list view to a detail view, then Siebel CRM preserves the query context, and Siebel CRM includes the records in the detail view.

CAUTION: It is recommended that you choose no more than 50 records. If you choose more than 50 records, then the report might fail. For more information, see [“Allowing Users to Choose The Records That a Report Contains” on page 106](#).

Caution About Running Reports with a Large Number of Records

It is strongly recommended that you run a query that filters records in the client before you run a report.

CAUTION: It is strongly recommended that you run a query that filters records in the client before you run a report. Running a query reduces the number of records that Siebel CRM includes in a report. Some views might include hundreds or even thousands of records. If you do not filter these records before you run the report, then the report might fail, and performance might degrade.

Entering a Custom Name

You can enter a custom name for the report in the Custom Name field instead of choosing a report from the Report Name list. If you do this, and then click Submit, then Siebel CRM creates a new report and uses the value you enter in the Custom Name field to set the name of this new report.

Monitoring Report Status

This topic describes how to monitor the status of reports that are currently running in the client. It does not describe how to monitor a scheduled report. For information, see [“Monitoring, Suspending, Viewing, and Deleting Scheduled Reports” on page 29](#).

To monitor report status

- 1 Log in to the client.
- 2 Navigate to the BI Publisher Reports screen, and then the My Reports view.

You can also click My Reports in the Run Report pane.

- 3 In the My Reports list, locate the report that you must monitor, and then examine the Status field using values from the following table.

Status	Description
In Progress	Indicates a report is currently running.
Success	Indicates a report successfully finished running.
Error	Indicates a report encountered an error while running.

A short delay might occur before the client displays a value for the status.

Viewing Report Output

This topic describes how to view details about a report that finished running. The report includes the state of the data that exists when you run the report. You can view reports that:

- You create
- Your team members create
- Anyone creates

Siebel CRM saves these reports in the Siebel File System. For information about how Siebel CRM determines if you can view a report, see [“How Siebel CRM Controls Access to Reports” on page 87](#).

To view report output

- 1 Log in to the client.
- 2 Navigate to the BI Publisher Reports screen, and then the Reports view.
You can also click My Reports in the Run Report pane.
- 3 In the Reports list, click a link in the Report Name field.

Siebel CRM displays the report according to your browser settings. For example, assume the Output Type is PDF, you use Google Chrome, and that you set up Google Chrome to download files to the following folder:

C: \Documents and Settings\user_name\My Documents\Downloads

In this example, Siebel CRM downloads the report file to the Downloads folder. To view the report output, you can use Windows Explorer to navigate to this folder, and then use Adobe Acrobat Reader to open the report file.

- 4 In the Reports view, examine fields in the Report Parameters form using values from the following table.

Field	Description
Bookmark	<p>Describes the query details, including the applet search expressions, business component search expressions, and sort specifications that are active when you run the report. For example:</p> <pre>' Account' . Search = " ([Type] LI KE ""Resi denti al *"")" ' Account' . Sort = "Name(ASCENDI NG), Locati on"</pre> <p>For more information about search expressions and sort specifications, see <i>Configuring Siebel Business Applications</i>.</p>
Description	A text entry field that allows you to include a description of the report.
Start Time End Time	The start and end times that indicate when Siebel CRM ran the report.
Error Message	Includes any error messages that occurred when Siebel CRM ran the report.
File Size	Describes the size of the output file, in kilobytes.
Query String	Displays the query you entered immediately before you ran the report.
Report Access	Allows you to specify other users who can access this report.
Tags	An editable field you can use to add more descriptive information. This field is for your documentation purposes. It does not effect configuration in any way.
Task Id	Identifies the unique task ID of the report server request. You can use this ID to locate information that is related to this report request if you examine a log.
View Mode	Identifies the view mode that Siebel CRM used when it created the report. The value in this field is useful for administrators and debugging.

Deleting Reports

This topic describes how to delete a report. It does not describe how to delete a scheduled report. For more information, see [“Scheduling Reports” on page 25](#).

To delete a report

- 1 Navigate to the BI Publisher Reports screen, and then the Reports view.

You can also click My Reports in the Run Report pane.

NOTE: Choose a report in the Reports list, and then click Delete.

You can delete any report that you create or that you can access. Siebel CRM removes the report from the client and from the Siebel File System.

Scheduling Reports

This topic describes how to schedule a report. Note the following:

- You can schedule a report in Siebel Open UI. You can also schedule a report that includes parameters. For more information, see [“Using Report Parameters to Filter Reports” on page 29](#).
- You can schedule a report only in a connected client. For more information, see [“How Siebel CRM Runs Reports in Connected Clients” on page 79](#).

To schedule a report

- 1 Log in to the client, and then navigate to a view where you must schedule a report.
- 2 Click the Reports button in the application toolbar.
- 3 In the Run Report pane, choose a report in the Report Name list.
- 4 Expand the Schedule section.

Some reports might not include a schedule. If the Run Report pane does not display the Schedule section, then you must choose another report in the Report Name list. If a report does not include a schedule, then you can configure Siebel CRM to add one. For more information, see [Chapter 7, “Customizing Siebel CRM Reports.”](#)

- 5 Set the schedule.

The elements that Siebel CRM displays in the Schedule section depends on the report that you choose in [Step 3](#). The report template that Siebel CRM uses for this report determines the elements that it displays. For more information, see [“Setting the Schedule for a Report” on page 26](#).

If you cannot access a report, then ask your administrator to assign the XMLP_SCHEDULER responsibility to you. For information, see [“How Siebel CRM Controls Access to Reports” on page 87](#).

- 6 (Optional) Enter a custom name.

For information, see [“Entering a Custom Name” on page 22](#).

- 7 Click Submit.

Siebel CRM displays a message that indicates that it scheduled the report. This message includes a job ID.

- 8 Monitor the report status.

For more information, see [“Monitoring Report Status” on page 22.](#)

- 9 View the report output.

For more information, see [“Viewing Report Output” on page 23.](#)

Setting the Schedule for a Report

This topic describes how to set the schedule for a report.

To set the schedule for a report

- 1 Click the Frequency list in the Schedule Section of the Run Report Pane.
- 2 Choose a value using information from the following table.

Value	Description
Run Immediately	Siebel CRM runs the report immediately when you click Submit.
Run Once	<p>Siebel CRM displays the Run Date element that allows you to specify the time, day, month, and year to run this report. You use the following elements to specify the time:</p> <ul style="list-style-type: none"> ■ Hour. Choose the hour of the day when Siebel CRM must run the report. ■ Minute. Choose the minutes after the hour when Siebel CRM must run the report. <p>For example, if you set Hour to 23, and Minute to 30, then Siebel CRM runs the report at 11:30 PM according to UTC (Coordinated Universal Time).</p> <p>If you click Submit, then Siebel CRM runs the report only one time according to the run date that you specify.</p>

Value	Description
Run Daily/Weekly	<p>Siebel CRM displays the following elements, and you can set a value for each of these elements. If you click Submit, then Siebel CRM runs the report according to the values that you specify:</p> <ul style="list-style-type: none"> ■ Run Time. Specify the hour and minutes after the hour when Siebel CRM must run the report. For example, if you choose 23 in the Run Time list (left), and then 30 in the Run Time list (right), then Siebel CRM runs the report at 11:30 PM according to the time zone that the Oracle BI Publisher Server uses. ■ Start Date. Set the day, month, and year when Siebel CRM must run the first instance of this report. The Start Date and End Date use the date format that the user preference specifies. ■ End Date. Set the day, month, and year when Siebel CRM must run the last instance of this report. ■ Days. Choose the day or days of the week when Siebel CRM must run this report. ■ Public. If you make this button active, then all users can view this report. ■ Save Data. If you make this button active, then Siebel CRM saves the unformatted report data in an XML file. ■ Save Output. If you make this button active, then Siebel CRM saves the formatted report output, such as a PDF file, and you can view this output in the Scheduled Reports view.
Run Monthly	<p>Siebel CRM displays the same elements that it displays for Run Daily/Weekly, except it does not display the Days element. It also displays the following elements. You can set a value for each of these elements. If you click Submit, then Siebel CRM runs the report according to the values that you specify:</p> <ul style="list-style-type: none"> ■ Months. Choose the month or months when Siebel CRM must run this report. ■ Dates. Enter the day, days, or a date range when Siebel CRM must run this report. For example, you enter the following value in the Dates field, and if you choose March for the month, then Siebel CRM will run the report on the second, third, and fourth days of March: <p style="text-align: center;">2-4</p> <p>You can also use a comma to chose nonconsecutive dates. For example, if you enter the following value in the Dates field, then Siebel CRM will run the report on the first, third, and fifth days of the month:</p> <p style="text-align: center;">1, 3, 5</p>

Example of Scheduling a Report

The example in this topic runs the By Sales Rep opportunity report in HTML format every Tuesday and Thursday at 10:15 a.m. for a two week period, starting on August 2-15, 2009.

To schedule the By Sales Rep Opportunity report

- 1 Navigate to the Opportunities screen, My Opportunities, and then the Opportunity list.
- 2 Run a query to filter the number of opportunity records.

For more information, see [“Caution About Running Reports with a Large Number of Records” on page 22.](#)

- 3 Click the Reports button in the application toolbar.
- 4 In the Run Report pane, set element values using information from the following table.

Element	Value
Report Name	By Sales Rep
Custom Name	My Opportunities Report
Output Type	PDF
Report Locale	English - United States

- 5 Expand the Schedule section, and then set the schedule using values from the following table.

Element	Value
Frequency	Run Daily/Weekly.
Run Time	Choose 23 in the left Run Time list. Choose 30 in the right Run Time list.
Start Date	Set the day, month, and year.
End Date	Set the day, month, and year.
Days	Choose the day or days of the week when Siebel CRM must run this report. Siebel CRM displays each day as not chosen, by default. You must choose each day that you require Siebel CRM to run the report.
Public	Make this button active.
Save Data	Make this button inactive.
Save Output	Make this button active.

- 6 Click Submit.

- 7 Monitor the report status and view the report output.

For more information, see [“Monitoring Report Status” on page 22](#) and [“Viewing Report Output” on page 23](#).

Monitoring, Suspending, Viewing, and Deleting Scheduled Reports

This topic describes how to monitor a report that is currently running, temporarily suspend a report that is currently running, view a report that finished running, and delete a report.

To monitor, suspend, view, and delete scheduled reports

- 1 Navigate to the BI Publisher Reports screen, and then the Scheduled Reports view.
- 2 Monitor the status in Oracle BI Publisher by doing one of the following:
 - Click the Schedules tab to view the report jobs you scheduled and their status. Drill down on a report to view detailed information about the report job.
 - Click the History tab to view a history of your report jobs.
- 3 (Optional) Suspend and resume a scheduled report:
 - a Click the Schedules tab.
 - b Click the check box next to a report, and then click Suspend or Resume.
- 4 (Optional) Delete a scheduled report:
 - a Navigate to the Scheduled Reports view.
 - b Click the check box next to the report you must delete, and then click Delete.

This step removes the report instance from the report list. It does not remove the report from the Oracle BI Publisher repository. For more information, see [“Deleting Scheduled Reports from the Oracle BI Publisher Server” on page 64](#).

Using Report Parameters to Filter Reports

NOTE: The example in this topic describes how to use a report parameter to filter a report in Siebel Call Center. A *report parameter* is a type of filter that refines the data that Siebel CRM includes in a report. It allows you to do the following work:

- Narrow the query when you run a report.
- Create different reports from the same report template.

Siebel Open UI supports report parameters starting with Siebel CRM versions 8.1.1.10 and 8.2.2.3. For more information, see [“Customizing Reports That Use Parameters” on page 107](#).

To use report parameters to filter reports

- 1 Log in to Siebel Call Center, and then navigate to a view that includes a report that includes report parameters.
For example, navigate to the Accounts list.
- 2 Run a query in the Accounts list to filter the number of account records that Siebel CRM includes in the report.
For more information, see [“Caution About Running Reports with a Large Number of Records” on page 22](#).
- 3 Click the Reports button in the application toolbar.
- 4 In the Run Report pane, in the Report Name list, choose Parameter.
- 5 Expand the Parameters section.
- 6 Set Param4 to February.
Siebel CRM gets these report parameters from the report template. For more information, see [“Adding Report Parameters to Report Templates” on page 108](#).
- 7 Click Submit.

The Oracle BI Publisher Server runs the report, and then Siebel CRM sends it to the client. If you include a report parameter, and if you also schedule a report to run at some point in the future, then Siebel CRM stores the values you set for the report in memory when you click Submit, and you cannot modify them. If you must modify these values after you click Submit, then you must modify them and then run the report again.

4

Integrating Oracle BI Publisher with Siebel CRM for Siebel Innovation Pack 2017

This chapter provides instructions for administrators on how to integrate Oracle Business Intelligence Publisher (Oracle BI Publisher) 11.1.1.9 or 12c with Siebel CRM for report generation for Siebel Innovation Pack 2017 for first-time installations as well as upgrades. It includes the following topics:

- [About Integrating Oracle BI Publisher with Siebel CRM on page 31](#)
- [Requirements for Using Siebel Reports for First-Time Installations on page 32](#)
- [Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations on page 33](#)
- [Preparing for Integration of Oracle BI Publisher with Siebel CRM on page 35](#)
- [Process of Installing Oracle BI Publisher for Integration with Siebel CRM on page 36](#)
- [Upgrading to the Latest Oracle BI Publisher Patch on page 41](#)
- [Configuring an Outbound Web Service for Siebel Reports on page 41](#)
- [Enabling Siebel Server Component Groups for Siebel Reports on page 42](#)
- [About Security and Authentication for Siebel Reports on page 43](#)
- [About Oracle Business Intelligence Enterprise Edition Coexistence with Siebel Reports on page 44](#)
- [Configuring Security and Authentication for Siebel Reports Using LDAP on page 48](#)
- [Configuring Siebel CRM for Integration with Oracle BI Publisher with Web Single Sign-On on page 50](#)
- [Adding an Explicit Reference to a Container URL for Oracle BI Publisher Server on page 54](#)
- [Implementing EAI File Handling Changes for Siebel Reports Generation on page 55](#)
- [Uploading Preconfigured Report Layout Templates to Oracle BI Publisher Server on page 56](#)
- [Process of Enabling and Configuring Report Scheduling on page 57](#)
- [About Upgrading Siebel Reports for Siebel Innovation Pack 2017 on page 59](#)

About Integrating Oracle BI Publisher with Siebel CRM

Oracle Business Intelligence Publisher (Oracle BI Publisher) is the reporting module for Siebel CRM. Siebel Reports integrates with Oracle BI Publisher for report execution and administration. Oracle WebLogic Server and Oracle BI Publisher Server are installed as part of the Oracle BI Publisher Installation.

You might need to install and configure Siebel CRM to use Oracle BI Publisher so that it can run reports. The tasks you perform depend on whether you must use Siebel Reports for the first time or if you are already using it. They also depend on the Siebel CRM version that your deployment uses. If you have an existing Siebel Reports environment, and if you must use the latest report features, then you must upgrade to the latest Siebel CRM Fix Packs 8.1.1.x or 8.2.2.x and do other configuration work for the Siebel application and Oracle BI Publisher.

Some Siebel Business Applications do not require a Siebel client. For example, Siebel Loyalty can use the Oracle BI Publisher libraries to run Oracle BI Publisher reports. You can also use a business service in the XMLP Report server component to request a report without using a Siebel client. This guide does not describe these usages.

For more information about how Siebel CRM integrates Siebel Reports and the architecture it uses, see [“About the Siebel Reports Architecture” on page 77](#).

Each new Siebel CRM release typically introduces new report features and functionality. However, before you can use these latest features for first-time installations, you must perform several postinstallation and configuration tasks. For a step-by-step view of these tasks, see [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations” on page 33](#). If you are upgrading from previous releases, you only need perform the instructions in [“About Upgrading Siebel Reports for Siebel Innovation Pack 2017” on page 59](#).

Requirements for Using Siebel Reports for First-Time Installations

The reporting module for Siebel CRM is Oracle Business Intelligence Publisher (Oracle BI Publisher). This topic provides the requirements you must meet before you can use Siebel Reports for the first time.

The following requirements must be met before using Siebel Reports:

- You must have the latest Siebel CRM version.

For information about supported Siebel CRM versions, see the Certifications tab on My Oracle Support. For information about Certifications, see article 1492194.1 (Article ID) on My Oracle Support. Some of the requirements include:

- A Siebel Server (or servers)
- A Siebel Web Client for connected mode
- A Siebel mobile client (MWC or DWC) for disconnected mode
- Siebel Tools
- (Optional) Application Deployment Manager (ADM)

You might want to use ADM to migrate reports from one Siebel environment to another (for example, migrating reports from the development to the production environment). For more information about ADM, see *Siebel Application Deployment Manager Guide*.

The following requirements are met as part of implementing instructions in this document:

- Java Development Kit (JDK) with the JAVA_HOME system environment variable pointing to the JDK. To find the JDK version applicable to your deployment, see the Certifications tab on My Oracle Support.
- Siebel Server components enabled:
 - Siebel Application Object Manager (AOM), for example, SCCObjMgr_enu for Siebel Call Center
 - EAI Object Manager
 - XMLP Report Server
 - (Optional) Workflow
- Oracle BI Publisher software
 - Oracle Business Intelligence Publisher
 - Oracle BI Publisher Add-In for Microsoft Word (also known as Oracle BI Publisher Desktop)

NOTE: You must enable the Workflow component if you plan to use workflow, the Reports Business Service, or the XMLP Purge Records workflow.

To find the version of Oracle BI Publisher software applicable to your deployment, see the Certifications tab on My Oracle Support.

Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations

[Figure 4 on page 34](#) provides a roadmap for integrating Oracle BI Publisher with Siebel CRM. This integration is a requirement for deploying Siebel Reports for the first time.

NOTE: It is recommended that you print this roadmap to use as a checklist as you work your way through the integration.

Explanation of Callouts

In Figure 4, the following conventions are used:

- A Siebel application view is first in the order (black box).
- A task follows a Siebel application view (green box).
- A subtask follows a task (brown box).

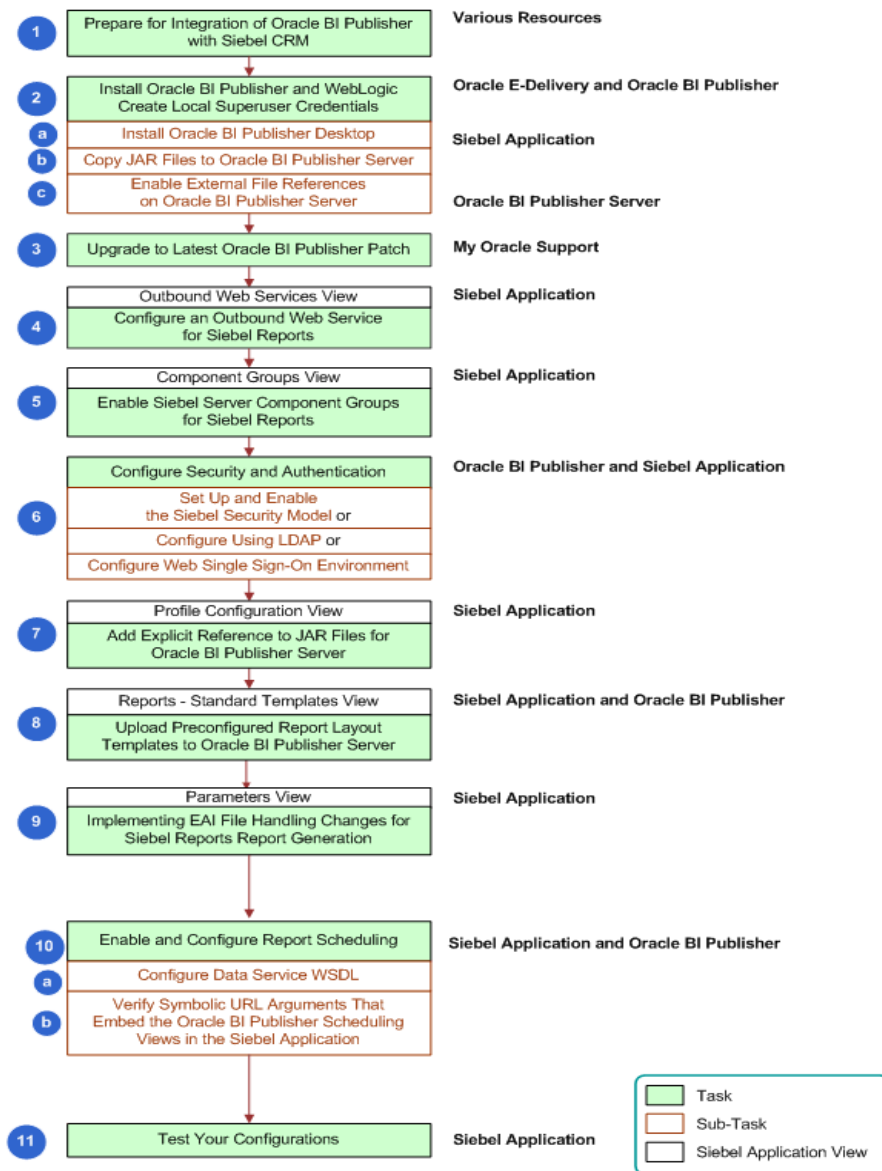


Figure 4. Roadmap for Integrating Oracle BI Publisher with Siebel CRM

Figure 4 illustrates the high-level tasks you perform to integrate Oracle BI Publisher with Siebel CRM if you are implementing Siebel Reports for the first time. Click the following links to navigate to detailed instructions for that topic.

- 1 [“Preparing for Integration of Oracle BI Publisher with Siebel CRM” on page 35](#)
- 2 [“Process of Installing Oracle BI Publisher for Integration with Siebel CRM” on page 36](#)
 - a [“Installing Oracle Business Intelligence Publisher Add-in for Microsoft Word” on page 38](#)
 - b [“Copying JAR Files to Oracle BI Publisher Server” on page 39](#)
 - c [“Enabling External File References on Oracle BI Publisher Server” on page 40](#)
- 3 [“Upgrading to the Latest Oracle BI Publisher Patch” on page 41](#)
- 4 [“Configuring an Outbound Web Service for Siebel Reports” on page 41](#)
- 5 [“Enabling Siebel Server Component Groups for Siebel Reports” on page 42](#)
- 6 [“About Configuring Security and Authentication for Siebel Reports” on page 45](#)
 - [“Setting Up and Enabling the Siebel Security Model for Siebel Reports” on page 46](#)
 - [“Configuring Security and Authentication for Siebel Reports Using LDAP” on page 48](#)
 - [“Configuring Siebel Reports in a Web Single Sign-On Environment” on page 49](#)
- 7 [“Adding an Explicit Reference to a Container URL for Oracle BI Publisher Server” on page 54](#)
- 8 [“Uploading Preconfigured Report Layout Templates to Oracle BI Publisher Server” on page 56](#)
- 9 [“Implementing EAI File Handling Changes for Siebel Reports Generation” on page 55](#)
- 10 [“Process of Enabling and Configuring Report Scheduling” on page 57](#)
 - a [“Configuring the Data Service WSDL for Oracle BI Publisher Server” on page 57](#)
 - b [“Verifying Symbolic URL Arguments That Embed the Oracle BI Publisher Scheduling Views in the Siebel Application” on page 58](#)

Preparing for Integration of Oracle BI Publisher with Siebel CRM

The reporting module for Siebel Reports is Oracle Business Intelligence Publisher (Oracle BI Publisher). To deploy Siebel Reports, you must install and configure both Siebel CRM and Oracle BI Publisher. This topic provides the necessary preparation tasks for a new deployment and supports the latest reports enhancements.

CAUTION: If you are upgrading from a previous Siebel Reports deployment, do not perform the steps described in this topic. Instead, see [“About Upgrading Siebel Reports for Siebel Innovation Pack 2017” on page 59](#)

This task is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations” on page 33](#).

To prepare for integration of Oracle BI Publisher with Siebel CRM

- 1 Make sure you have met the requirements for Siebel Reports.
For the requirements, see [“Requirements for Using Siebel Reports for First-Time Installations”](#) on page 32.
- 2 Install the appropriate Java Development Kit (JDK) version and point the JAVA_HOME system environment variable to the JDK, if you have not already done so.
To find the version of JDK applicable to your deployment, see the Certifications tab on My Oracle Support. For information about Oracle BI Publisher 11g, see 1496816.1 (Article ID) on My Oracle Support for specific, detailed integration information.
- 3 Before using Siebel Reports, make sure that the Siebel Server is running and the Siebel clients and Siebel Tools can connect to the server data source.

Process of Installing Oracle BI Publisher for Integration with Siebel CRM

This topic describes how to install Oracle Business Intelligence Publisher (Oracle BI Publisher) as a first-time installation for integration with Siebel CRM.

NOTE: Do not use this process if you are upgrading from a previous deployment of Siebel Reports.

This process is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations”](#) on page 33.

To install Oracle BI Publisher for integration with Siebel CRM, perform the following tasks:

- 1 [“Installing Oracle Business Intelligence Publisher Enterprise”](#) on page 37
For information about which versions of this software to use for your specific deployment, see the Certifications tab on My Oracle Support.
- 2 [“Installing Oracle Business Intelligence Publisher Add-in for Microsoft Word”](#) on page 38
For information about which versions of this software to use for your specific deployment, see the Certifications tab on My Oracle Support.
- 3 [“Copying JAR Files to Oracle BI Publisher Server”](#) on page 39
- 4 [“Enabling External File References on Oracle BI Publisher Server”](#) on page 40
- 5 (Optional) [“Starting and Stopping Oracle WebLogic Server and Oracle BI Publisher”](#) on page 40
- 6 [“Upgrading to the Latest Oracle BI Publisher Patch”](#) on page 41

NOTE: This task is applicable to both first-time Siebel Reports deployments as well as upgrades.

Installing Oracle Business Intelligence Publisher Enterprise

Use the following procedure to install Oracle BI Publisher Enterprise for integration with Siebel CRM and Siebel Reports.

This task is a step in “[Process of Installing Oracle BI Publisher for Integration with Siebel CRM](#)” on page 36.

To install Oracle BI Publisher Enterprise

- 1 Log in to Oracle Software Delivery Cloud available at:
<http://edelivery.oracle.com>
- 2 Enter the export validation information, accept the license terms and export restrictions, and then click Continue.
- 3 Select Oracle Business Intelligence as the product pack, and an operating system (for example, Microsoft Windows [32-bit]). Then click Go.
NOTE: You must select an operating system compatible with the computer on which Oracle BI Publisher Server is installed. For information about which platforms are supported, see Oracle® Business Intelligence Publisher Certification information at
http://docs.oracle.com/cd/E12844_01/doc/bip.1013/e12692/toc.htm#sthref2
- 4 Select the appropriate Oracle Business Intelligence Media Pack, for example, you might select Oracle Business Intelligence (11g) Media Pack for Microsoft Windows (32-bit).
- 5 Download the appropriate Oracle BI Publisher installer for your deployment, for example, version 11.1.1.9 or 12c. For information about which versions of this software to use for your specific deployment, see the Certifications tab on My Oracle Support.
NOTE: Typically administrators download and install Oracle Business Intelligence Publisher Desktop (also known as Oracle Business Intelligence Publisher Add-in for Microsoft Word) during the Oracle BI Publisher Enterprise installation, however, you can install it any time. For information on installing this desktop software, see “[Installing Oracle Business Intelligence Publisher Add-in for Microsoft Word](#)” on page 38.
- 6 Install Oracle BI Publisher by doing the following:
CAUTION: You must extract the .zip file to a directory whose name does not contain spaces. If the directory name contains spaces, the installation fails.
 - a Decompress the installer into a directory of your choice.
 - b Follow the instructions about installing Oracle BI Publisher on Oracle Application Server for Oracle WebLogic Server.
 - c Verify the installation by making sure you have an Oracle - BIPHomeX entry in the Start Programs menu, where X is an incremental number depending on the number of Oracle installations you have.
- 7 Enable Superuser for Oracle BI Publisher Server by doing the following:

- a Start Oracle BI Publisher (for example, <http://BIPServerHostName:9704/xmlpserver>), and then log in to Oracle BI Publisher Server with administrator credentials.

NOTE: The URL for starting Oracle BI Publisher resides in the `BI_Publisher_readme.txt` file located in the Oracle BI Publisher installation folder. Administrator credentials are case sensitive.

- b Click the Admin tab, and then select Security Configuration (in the Security Center section).
- c Check the Enable Local Superuser check box, enter a Superuser name and password of your choice, and then click Apply.

After creating a local superuser, administrators can log in directly to Oracle BI Publisher Server when the Siebel Server is not running.

Installing Oracle Business Intelligence Publisher Add-in for Microsoft Word

Use the following procedure to install Oracle Business Intelligence Publisher Add-in for Microsoft Word (also known as Oracle BI Publisher Desktop) for integration with Siebel CRM and Siebel Reports. For information about which versions of this software to use for your specific deployment, see the Certifications tab on My Oracle Support.

NOTE: Ensure that applicable users install Oracle BI Publisher Desktop on a computer where Microsoft Word for Windows is installed.

This task is a step in “[Process of Installing Oracle BI Publisher for Integration with Siebel CRM](#)” on page 36.

To install Oracle BI Publisher Add-in for Microsoft Word

- 1 Log in to Oracle Software Delivery Cloud available at:

<http://edelivery.oracle.com>

NOTE: This software is listed on Oracle Software Delivery Cloud as Oracle Business Intelligence Publisher Desktop.

- 2 Enter the export validation information, accept the license terms and export restrictions, and then click Continue.
- 3 Select Oracle Business Intelligence as the product pack, and an operating system (for example, Microsoft Windows [32-bit]). Then click Go.

NOTE: You must select an operating system compatible with the computer on which Oracle BI Publisher Server is installed.

- 4 Select the appropriate Oracle Business Intelligence product pack, and then click Continue.

For example, you might select Oracle Business Intelligence (11.1.1.7) product pack for Microsoft Windows (32-bit).

- 5 Download the appropriate Oracle BI Publisher installer for your deployment.

NOTE: Typically administrators download and install Oracle Business Intelligence Desktop during the Oracle BI Publisher Enterprise installation, however, you can install it any time.

- 6 Install Oracle Business Intelligence Publisher Add-in for Microsoft Word by doing the following:
 - a Decompress the appropriate .zip file according to the Media Pack you chose in [Step 3](#).

CAUTION: You must extract the .zip file to a directory whose name does not contain spaces. If the directory name contains spaces, the installation fails.
 - b Run the installer.
 - c Verify the installation by making sure you have an Oracle BI Publisher Desktop application in the Start Programs menu.

Copying JAR Files to Oracle BI Publisher Server

This topic describes how to add custom Siebel Java archive (JAR) files to Oracle BI Publisher Server. To successfully generate reports, you must copy the custom Siebel Java classes to the corresponding directory on Oracle Business Intelligence Publisher (Oracle BI Publisher) Server. By default, these Java classes are not available on Oracle BI Publisher Server.

This task is a step in [“Process of Installing Oracle BI Publisher for Integration with Siebel CRM”](#) on [page 36](#).

Before You Begin. The administrator must create the Classes folder in the `<InstallLocation>\user_projects\domains\bifoundation_domain` directory.

To copy JAR files to Oracle BI Publisher Server

- 1 Copy the following files from the `SI_EBSRVR_ROOT\CLASSES` directory to the `<InstallLocation>\user_projects\domains\bifoundation_domain\Classes` directory where `<InstallLocation>` is the install path for BI Publisher 11g.

- Siebel.jar
- SiebelXMLP.jar
- SiebelCustomXMLP.jar
- SiebelCustomXMLP_SIA.jar
- XSLFunctions.jar
- (Optional) InkToolsLib.jar
- (Optional) iSignBmp.jar
- SignMigration.jar

NOTE: Typically, `<InstallLocation>\user_projects\domains\bifoundation_domain\Classes` is the Oracle BI Publisher installation directory for Windows. The `InkToolsLib.jar`, `iSignBmp.jar`, and `SignMigration.jar` files are required whenever required signatures must appear in a Siebel report using the Siebel signature capture feature.

- 2 If you have created custom Java function libraries, then you must copy those JAR files as well.

Enabling External File References on Oracle BI Publisher Server

This topic provides instructions for enabling a run-time configuration property on Oracle BI Publisher Server for report generation. The preconfigured report templates that ship with Siebel CRM reference certain functions, such as psfn functions, which are contained in a separate XLSFunctions.jar file. For these psfn functions to be available to the templates during report generation, an external library file on Oracle BI Publisher Server must be referenced from Siebel.

This task is a step in [“Process of Installing Oracle BI Publisher for Integration with Siebel CRM”](#) on page 36.

To enable external file references on Oracle BI Publisher Server

- 1 Log on to Oracle BI Publisher Server with administrator privileges, if you have not already done so.
- 2 Click the Admin tab, and then select Properties under Runtime Configuration.
- 3 Change the default value for the Disable External Reference attribute to FALSE, and then click Apply.
- 4 Restart Oracle BI Publisher Server for the changes take effect.

For instructions, see [“Starting and Stopping Oracle WebLogic Server and Oracle BI Publisher”](#) on page 40.

Starting and Stopping Oracle WebLogic Server and Oracle BI Publisher

There are many instances where you must start, stop, or restart Oracle WebLogic Server and Oracle BI Publisher.

Starting Oracle WebLogic Server and Oracle BI Publisher

Use the following procedure to start Oracle WebLogic Server and Oracle BI Publisher.

To start Oracle WebLogic Server and Oracle BI Publisher

- 1 Click Start, Programs, Oracle Business Intelligence, and then Start BI Services.

A command window appears.

NOTE: You must leave this command window open while Oracle BI Publisher Server is running.

- 2 Confirm Oracle BI Publisher Server is started.

When the server is started, a statement appears in the command window indicating that the WebLogic is initialized.

Stopping Oracle WebLogic Server and Oracle BI Publisher

Use the following procedure to stop Oracle WebLogic Server and Oracle BI Publisher.

To stop Oracle WebLogic Server and Oracle BI Publisher

- 1 Click Start, Programs, Oracle Business Intelligence.
- 2 Click Stop BI Publisher.

Restarting Oracle WebLogic Server and Oracle BI Publisher

Use the following procedure to restart Oracle WebLogic Server and Oracle BI Publisher.

To restart Oracle WebLogic Server and Oracle BI Publisher

- 1 Click Start, Programs, Oracle Business Intelligence
- 2 Click Stop BI Publisher.
- 3 Click Start BI Publisher.

Upgrading to the Latest Oracle BI Publisher Patch

To use the latest reporting features, it is recommended that you upgrade to the latest Oracle Business Intelligence Publisher (Oracle BI Publisher) patch release. For more information about Oracle BI Publisher patches, see 797057.1 (Article ID) on My Oracle Support. For information about installing Oracle BI Publisher patches in a Siebel environment, see 1262585.1 (Article ID) on My Oracle Support.

This task is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations”](#) on page 33.

Configuring an Outbound Web Service for Siebel Reports

This topic provides instructions for configuring an outbound Web service to integrate Siebel applications with Oracle BI Publisher Server for running Siebel Reports. It ensures that the service port address corresponds to Oracle BI Publisher Server.

This task is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations”](#) on page 33.

To configure an outbound Web service for Siebel Reports

- 1 Navigate to the Administration - Web Services screen, then the Outbound Web Services view.
- 2 In the Name field, query for PublicReportServiceService.
- 3 Change or verify that the service port address corresponds with Oracle BI Publisher Server.

The format is:

`http://WebServerHostName:port/xmlpserver/services/PublicReportService_v11`

where:

- *WebServerHostName* is the full path for Oracle BI Publisher Server
- *port* is the port on which Oracle BI Publisher Server resides

For example, the address might be:

`http://BI PServerHostName:7001/xmlpserver/services/PublicReportService_v11`

- 4 Clear the Siebel Server cache to make sure that the changes are propagated to run-time memory.

Enabling Siebel Server Component Groups for Siebel Reports

Before you can work with Siebel Reports, you must enable several Siebel Server component groups.

This task is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations”](#) on page 33.

To enable server component groups for Siebel Reports

- 1 Navigate to the Administration - Server Configuration screen, Enterprises, then the Component Groups view.

- 2 Enable the following component groups on the Siebel Server:

Component Group	Component Name	Component Alias
EAI	EAI Object Manager (ENU)	EAIObjMgr_enu
XMLPReport	XMLP Report Server	XMLPReportServer
(Optional) Workflow Management	Workflow Process Manager	WfProcMgr

NOTE: Enable the Workflow Management component group if you plan to use workflow, the Reports Business Service, or the XMLP Purge Records workflow. Additionally, the XMLP Report Server component might already be enabled depending on how you configured your original Siebel Server installation.

For information about enabling Siebel Server components, see *Siebel System Administration Guide*.

About Security and Authentication for Siebel Reports

There are several options to choose from for configuring security and authentication for Siebel Reports: Siebel Security Model, LDAP, Web single sign-on, and Oracle BI Publisher Security. This topic provides a high-level description of each option and provides links to the instructions for implementing each option.

Siebel Security Model

When the Oracle Business Intelligence Publisher (Oracle BI Publisher) Server receives a request from a Siebel user to perform report functionality on the server, it queries the Siebel application to retrieve the user privileges associated with that particular user. This retrieval is accomplished by invoking a Web service that is exposed by the Siebel application.

The Siebel Security model retrieves authentication information through the Siebel EAI Object Manager (EAIObjMgr) and authenticates logins for report users against the Siebel Database one time only, then a session token is provided for Oracle BI Publisher Server authentication. Because every user executing reports has their own credentials and roles, users have private visibility on Oracle BI Publisher Server when scheduling reports.

The reporting responsibilities in the Siebel application correspond to roles and permissions on Oracle BI Publisher Server. The names of these responsibilities must exactly match the corresponding set of roles on Oracle BI Publisher Server.

NOTE: It is recommended that you use this model because it was designed specifically for use with Siebel Reports.

LDAP

In an implementation using LDAP authentication, an LDAP security adapter provided with Siebel CRM authenticates users against an LDAP-compliant directory. The directory stores information that is required to allow users to connect to the database and Oracle BI Publisher Server and this information is retrieved by the security adapter when users try to run a report. When users log into a Siebel application that uses LDAP authentication, they are already authenticated for the current session.

Web Single Sign-On Authentication

This model allows you to use a nonstandard security model, such as Siteminder or some other type of custom single sign-on.

Oracle BI Publisher Security

When implementing Oracle BI Publisher Security authentication for Siebel user access to Oracle BI Publisher Server, all connections between the Siebel application and Oracle BI Publisher Server are made using the same generic user account. This generic user account is useful for certain security topologies where multiple applications and authentication directories are used with a single shared Oracle BI Publisher Server instance.

NOTE: The Oracle BI Publisher security model does not enable private scheduled reports. All scheduled reports are considered public. For more information on Oracle BI Publisher Security, see *Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher*.

For more information about Oracle BI Publisher Security, see the Defining a Security Model topic in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide* at

http://docs.oracle.com/cd/E10415_01/doc/bi.1013/e12188/T421739T475591.htm

About Oracle Business Intelligence Enterprise Edition Coexistence with Siebel Reports

Oracle Business Intelligence Enterprise Edition (OBIEE) is a comprehensive business intelligence platform that delivers a full range of analytic and reporting capabilities. If you set up Siebel CRM with Oracle BI Publisher for Siebel Reports to use an existing Oracle BI Publisher enterprise server that is part of an OBIEE server, then you have two security options to choose from:

- LDAP security model (recommended)

For LDAP security information relating to Siebel Reports, see ["Configuring Security and Authentication for Siebel Reports Using LDAP"](#) on page 48.

- Oracle BI Server Security model

For the Oracle BI Server Security model, you must define all users and those users must belong to designated groups in OBIEE. For more information, see *Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher*.

About Configuring Security and Authentication for Siebel Reports

Authentication between the Siebel application and Oracle BI Publisher Server performs the following:

- 1 Checks that a user exists in the directory as follows:
 - For LDAP, checks for a user in the directory.
 - For Siebel Security, checks by way of the endpoint that a user exists in the Siebel application.
 - For Oracle BI Publisher Security, there is no user directory check. The generic authentication credentials are checked against a user defined in Oracle BI Publisher.
 - For Oracle BI Publisher Security, responsibilities are not checked, instead the same generic user account is used for the connection to Oracle BI Publisher Server.

NOTE: For more information on Oracle BI Publisher Security, see *Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher*.
- 2 Checks that a user has certain roles depending on what action they are taking:
 - For LDAP, groups are created that relate to the Oracle BI Publisher roles. Users in the LDAP directory have these certain groups assigned to them. Oracle BI Publisher authentication to LDAP directory checks for a user if they have a certain group.
 - For Siebel Security, each user has certain XMLP responsibilities. Oracle BI Publisher authentication to the Siebel application, by way of an endpoint inbound Web service call, checks that a user has the correct responsibilities that relate to the Oracle BI Publisher roles.
- 3 Checks Siebel XMLP responsibilities that control the Siebel user interface for Reports menu restrictions (for example, XMLP_SCHEDULER responsibility for the scheduling option).

Configuring Security and Authentication for Siebel Reports

This task is a step in ["Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations"](#) on page 33. For more information about security and authentication for Siebel Reports, see ["About Security and Authentication for Siebel Reports"](#) on page 43.

To configure security and authentication for Siebel Reports

- Perform one of the following:
 - ["Setting Up and Enabling the Siebel Security Model for Siebel Reports"](#) on page 46
 - ["Setting Responsibilities in the Siebel Application"](#) on page 47
 - ["Configuring Siebel Reports in a Web Single Sign-On Environment"](#) on page 49
 - ["Configuring Security and Authentication for Siebel Reports Using LDAP"](#) on page 48

Setting Up and Enabling the Siebel Security Model for Siebel Reports

This topic describes how to implement the Siebel Security Model to authenticate user access to Oracle BI Publisher Server in a Siebel Reports connected environment. The Siebel Security Model for Siebel Reports supports SOAP header-based and URL-based authentication. By default, SOAP header-based authentication is enabled.

This task is a step in [“Configuring Security and Authentication for Siebel Reports”](#) on page 45.

To set up and enable the Siebel Security Model for Siebel Reports

- 1 Log in to the Siebel application with administrative privileges.
- 2 Navigate to the Administration - Web Services screen, then the Inbound Web Services view.
- 3 In the Name field in the Inbound Web Services list, query for BIPSiebelSecurityWS.
- 4 (SOAP header-based authentication only) Change the host, port, and locale of the Siebel Server by doing the following, and then clear the Siebel Server cache:

```
http://<server>:<port>/siebel/app/eai/
```

```
Lang?SWEEExtSource=SecureWebService&SWEEExtCmd=Execute&WSSOAP=1
```

where:

- *Server* is the Siebel Server name on which the Web service resides.
- *Port* is the https redirection port
- *Lang* is the three-letter designator for the locale of the Siebel Server.

For example, you might replace the existing host address with:

```
http://computer.example.com:16690/siebel/app/eai/  
enu?SWEEExtSource=SecureWebService&SWEEExtCmd=Execute&WSSOAP=1
```

- 5 Enable the Siebel Security Model by doing the following:
 - a Make sure Oracle BI Publisher is started.
 - b Log in to the Oracle BI Publisher Enterprise application with administrator privilege.
 - c Click the Admin tab, and then select Security Configuration (under Security Center).
 - d In the Security Model section, perform the following, and then click Apply:
 - Set the model to Siebel Security
 - Enter the host name and port as the Web Service Endpoint URL

NOTE: The address of the Web Service Endpoint parameter in Oracle BI Publisher Server is the same address as the inbound Web service used by the Siebel Server.

- ❑ Enter the Siebel administrator credentials

NOTE: Siebel administrator credentials are necessary for the Oracle BI Publisher application to log into the Siebel EAI object manager. These credentials are case sensitive and must match the Siebel database specifications.

- 6 Restart Oracle BI Publisher Server.

For information about restarting Oracle BI Publisher Server, see [“Restarting Oracle WebLogic Server and Oracle BI Publisher” on page 41](#).

- 7 Verify that the Siebel Security Model has been implemented by doing the following:

- a Start the Oracle BI Publisher Enterprise application (from the Start Menu, choose Start BI Publisher).
- b Log in to the Oracle BI Publisher Enterprise application using the Siebel administrator credentials.

The Siebel Security Model was successfully implemented if you are able to login using the Siebel credentials.

Setting Responsibilities in the Siebel Application

This topic describes how to set responsibilities in the Siebel application.

This task is a step in [“Configuring Security and Authentication for Siebel Reports” on page 45](#).

To set responsibilities in the Siebel application

- 1 Define seven new responsibilities in the Siebel application.

Navigate to the Administration - Application screen, then the Responsibilities view to add the following seven new responsibilities:

- ❑ XMLP_ADMIN: Assign this responsibility to an Oracle BI Publisher Server administrator with no access limitations.
- ❑ XMLP_DEVELOPER: Assign this responsibility to allow for uploading reports from the Oracle BI Publisher Server to the Siebel application.
- ❑ XMLP_SCHEDULER: Assign this responsibility to allow for scheduling of reports.
- ❑ XMLP_SIEBEL_GUEST: Assign this responsibility to limit access to reports. This responsibility only allows for generation, viewing, and deletion of user’s own reports.
- ❑ XMLP_ADMIN_USER: Assign this responsibility to an administrative user.
- ❑ XMLP_DEVELOPER_USER: Assign this responsibility to a development user.
- ❑ XMLP_SCHEDULER_USER: Assign this responsibility to a user who uses scheduled reports.

- 2 Associate the XMLP_ADMIN and XMLP_ADMIN_USER responsibility with the following views:

- ❑ All Report Output List View
- ❑ All Report Template Across Organization View

- ❑ All Report Template View
 - ❑ Manager's Report Output List View
 - ❑ Manager's Report Template View
 - ❑ Report Job List View
 - ❑ Report Output List View
 - ❑ Report Standard Template Registration Admin View
 - ❑ Report Template Registration Admin View
- 3 Associate the XMLP_SCHEDULER and XMLP_SCHEDULER_USER responsibility with the Report Job List View view.

This association allows users to view their scheduled reports in the My Jobs view in the Siebel application, which is an embedded user interface to Oracle BI Publisher.

Configuring Security and Authentication for Siebel Reports Using LDAP

This topic describes how to implement LDAP security adapter authentication to manage user access to Oracle BI Publisher Server in a Siebel Reports connected environment.

When implementing LDAP authentication for Siebel user access to Oracle BI Publisher Server, you implement LDAP authentication for your Siebel application, and you must use the same LDAP directory for Oracle BI Publisher Server.

This task is a step in [“Configuring Security and Authentication for Siebel Reports”](#) on page 45.

To configure security and authentication for Siebel Reports using LDAP

- Follow the instructions in 1103065.1 (Article ID) on My Oracle Support.

For more information on implementing LDAP for Oracle BI Publisher, refer to the Oracle BI Publisher documentation. For more information on setting up an LDAP directory, see the topic on setting up the LDAP or ADS Directory in *Siebel Security Guide*.

Example of Configuring Security and Authentication for Siebel Reports Using LDAP

This topic gives one example of an LDAP directory configuration that can be used with a Siebel application to Oracle BI Publisher integration. You might use this feature differently, depending on your business model.

NOTE: Make sure you have configured both the Siebel application and Oracle BI Publisher so that both are pointing to the appropriate LDAP directory.

To configure security and authentication for Siebel Reports using LDAP

- 1 In the Siebel application, add the following responsibilities:
 - XMLP_ADMIN_USER
 - XMLP_DEVELOPER_USER
 - XMLP_SCHEDULER_USER
- 2 In the LDAP server, create the following six groups:
 - XMLP_ADMIN
 - XMLP_DEVELOPER
 - SIEBEL_DEV
 - SIEBEL_ADMIN
 - SIEBEL_SCHEDULER
 - XMLP_SCHEDULER
- 3 Associate members to the groups.
- 4 Associate the LDAP users to relevant members.

These tasks ensure that when Oracle BI Publisher Server authenticates with the LDAP directory, the correct roles are used for each user and limits the reporting actions that a user can perform.

Configuring Siebel Reports in a Web Single Sign-On Environment

This task is a step in [“Configuring Security and Authentication for Siebel Reports” on page 45](#).

This topic describes the configuration tasks you must perform to configure Siebel CRM and Oracle Business Intelligence Publisher (Oracle BI Publisher) in a Web Single Sign-On (SSO) environment. Oracle BI Publisher is the reporting module for Siebel CRM. Siebel Reports integrates with Oracle BI Publisher to run and administer reports.

For information on configuring Siebel CRM and Oracle BI Publisher for Web SSO authentication, see the following topics:

- [“Configuring Siebel CRM for Integration with Oracle BI Publisher with Web Single Sign-On” on page 50](#)
- [“Configuring Oracle BI Publisher for Integration with Siebel CRM with Web Single Sign-On” on page 51](#)
- [“Enabling Reports Scheduling with Web Single Sign-On” on page 52](#)
- [“Enabling Transport Layer Security for Oracle BI Publisher Running on Oracle WebLogic Server” on page 53](#)

Configuring Siebel CRM for Integration with Oracle BI Publisher with Web Single Sign-On

This topic lists the configuration tasks you must perform for your Siebel application so that it can integrate with Oracle BI Publisher when Web Single Sign-On (SSO) authentication is implemented.

To configure Siebel CRM for BI Publisher integration in a Web SSO environment

- 1 For the Security Adapter Profile (LDAP Security Adapter profile) that is used for authentication and Web SSO, specify parameter values as shown in the following table.

Parameter Name	Value
Single Sign On	True
Trust Token	<i>password</i> This is the value of the TrustToken parameter used for SSO in the Application Interface profile.

- 2 For the server components listed in the following table, specify values for the parameters shown. Specify values for the LDAP security adapter.

Server Component	Parameter	Value
Application Object Manager and EAI Object Manager	Security Adapter Name	LDAPSecAdpt
	Security Adapter Mode	LDAP
	Username	<i>LDAP_USER_ID</i>
	Password	<i>password</i> The password associated with the <i>LDAP_USER_ID</i>
XMLP Report Server	Security Adapter Name	LDAPSecAdpt
	Security Adapter Mode	LDAP
	Username	<i>LDAP_USER_ID</i>
	Password	<i>password</i> This is the value of the TrustToken parameter used for SSO in the Application Interface profile.

NOTE: The *LDAP_USER_ID* values you specify must be an LDAP user who has a Siebel employee record, for example, AnonUserName, in the Application Interface profile.

- 3 In the [/eai_lang] section of the Application Interface profile, remove the following parameters:

Parameter Name	Value
Single Sign On	False
Trust Token	<i>TrustToken_Value</i>
UserSpec	<i>HTTP Header Variable</i>
User SpecSource	Header

TIP: Ensure that the redirection to SSO server for the Application Interface profile is not set up on the Siebel Application Interface.

NOTE: The Siteminder ERP Agent for Siebel (also known as Web Agent) and the Siteminder Policy Server is still used to obtain the User Identity in the form of a HTTP header variable called SIEBELUSER and SSO Authentication Ticket.

- 4 Restart the Siebel Server, and the Application Interface.
- 5 When the services are started, verify that the Application Object Manager, EAI Object Manager, and XMLP Report Server components are online.

If any of these services are unavailable, create a service request (SR) on My Oracle Support. Alternatively, you can phone Oracle Global Customer Support directly to create a SR or get a status update on your current SR. Support phone numbers are listed on My Oracle Support.

Configuring Oracle BI Publisher for Integration with Siebel CRM with Web Single Sign-On

This topic describes how to configure Oracle BI Publisher to integrate with Siebel CRM when Web Single Sign-On (SSO) authentication is implemented.

To configure Oracle BI Publisher for Siebel CRM integration in a Web SSO environment

- 1 Log into the Oracle BI Publisher Server with administrator credentials.
- 2 Click the Admin tab, then select Security Configuration in the Security Center section.
- 3 Change the value of the Administrator Password parameter for the Siebel Security Model to specify the value of the Trust Token (in clear text) specified for Web SSO in the Application Interface profile.
- 4 Restart the Oracle BI Publisher WebLogic server.

NOTE: After the Administrator Password parameter is set to specify the value of the Trust Token, any Siebel user who wants to log into the Oracle BI Publisher Server must enter the Trust Token value as the password.

Enabling Reports Scheduling with Web Single Sign-On

This topic describes how to enable Siebel Reports scheduling when Web Single Sign-On (SSO) authentication is implemented for Siebel CRM and when the Siebel Security Model is implemented for Siebel Reports.

Oracle BI Publisher issues an inbound Web service call (BIPDataService) to retrieve data from the Siebel application when reports are scheduled and executed. During this process, report users are authenticated against the EAI Application Object Manager. You must, therefore, use a non-SSO security adapter for reports scheduling.

To enable Siebel Reports scheduling when Web SSO is implemented

- 1 Create a new custom Siebel Server component based on the EAI Object Manager component, and name the new component BIP EAI Object Manager.

For information about creating custom Siebel Server component definitions, see *Siebel System Administration Guide*.

- 2 Create a new Siebel enterprise profile (named subsystem) by copying the security adapter profile used by the Application Object Manager. Do the following:

- If the Siebel application is using the ADSISecAdpt security adapter profile, create a copy of the profile and name it ADSISecAdpt_NoSSO.

For information about creating Siebel Enterprise Server named subsystems, see *Siebel System Administration Guide*.

- 3 Set the SSO profile parameter for the new security adapter profile you created in [Step 2](#) to False.
- 4 For the BIP EAI Object Manager component you created in [Step 1](#), specify values for the parameters shown in the following tables:

Parameter	Value (LDAP Authentication)
Security Adapter Name	LDAPSecAdpt_NoSSO
Security Adapter Mode	LDAP

- 5 Synchronize the new component definitions, then restart the Siebel Server and the Siebel Gateway services.

For information about synchronizing components on a Siebel Enterprise Server, see *Siebel System Administration Guide*.

- 6 Create a new application as part of the Application Interface profile and do the following:

- a Add the following basic information parameters:

Application Name = bi peai

Language =<lang>

Object Manager= EAI Object Manager(ENU)

Configure EAI HTTP Inbound Transport = TRUE

- b Add the following enhanced authentication parameters:
 - Anonymous User Name : *<Guest Login>*
 - Anonymous User Password: *<Guest Password>*
- 7 Create a second application as part of the Application Interface profile and do the following:
 - a Add the following basic information parameters:
 - Application Name = eai_anon
 - Language =*<lang>*
 - Object Manager= EAI Object Manager(ENU)
 - Configure EAI HTTP Inbound Transport = TRUE
 - Configure Anonymous Pool = TRUE
 - Anonymous Pool Size = *<Pool Size>*
 - b Add the following enhanced authentication parameters:
 - Anonymous User Name : *<Guest Login>*
 - Anonymous User Password: *<Guest Password>*
- 8 Launch the Siebel Web Client and log into the Siebel application as a Siebel administrator.
- 9 Navigate to the Administration - Web Services screen, then the Inbound Web Services view.
- 10 In the Name field of the Inbound Web Services list, query for BIPDataService.
- 11 In the address URL for the BIPDataService, change the value *eai_lang* to *eai_anon_lang*. For example:

```
http://SiebelWebServerName/eai_anon_lang/  
start.swe?SWExtSource=WebService&SWExtCmd=Execute&WSSOAP=1
```
- 12 Click the Generate WSDL button to generate a WSDL file, then save the file with the name `dataservice.wsdl`.
- 13 Copy the `dataservice.wsdl` file to the Oracle BI Publisher home directory. By default, this is the `<root dir>\user_projects\domains\bi_foundation_domain` directory on the Oracle BI Publisher server.
- 14 Restart the Oracle BI Publisher WebLogic server.

Enabling Transport Layer Security for Oracle BI Publisher Running on Oracle WebLogic Server

You must enable Transport Layer Security (TLS) for Web applications, such as Oracle BI Publisher, running on Oracle WebLogic server.

NOTE: Oracle BI Publisher does not control TLS/SSL. Oracle BI Publisher runs on Oracle WebLogic server and depends on the TLS/SSL environment used by Oracle WebLogic server.

To enable TLS for Oracle BI Publisher running on Oracle WebLogic server

- 1 Log in to the Oracle WebLogic server console.
- 2 Click <Domain>, click Environment, click Servers, and then <Server>.
- 3 Under Configuration and General, select the SSL Listen Port Enabled check box.
- 4 Select the SSL tab, click Advanced, and then select the Use JSSE SSL check box.

Choosing to use Java Secure Socket Extension (JSSE) ensures that Oracle WebLogic v10 and v11 uses the TLS features of Java, instead of any existing SSL implementation.

NOTE: The WebLogic server's internal SSL implementation is not compatible with current TLS implementations in modern browsers.

Oracle WebLogic server v12.2.1.0.0 uses JSSE by default and does not provide the option to switch back to an SSL implementation.

- 5 Restart Oracle WebLogic server for the changes to take effect.
- 6 To force the use of TLS v1.2, do the following:
 - a Open the setDomainEnv.sh file.
 - b In the WebLogic Startup parameter, set Dweblogic.security.SSL.protocolVersion to TLSv1.2:
Dweblogic.security.SSL.protocolVersion=TLSv1.2

Once this is done, any clients that do not support TLS v1.2 will be rejected.

Adding an Explicit Reference to a Container URL for Oracle BI Publisher Server

You must add an explicit reference to a container URL for Oracle BI Publisher Server to upload templates and run immediate reports or scheduled reports. This is because when a report is scheduled, a new .xdo file is sent to Oracle BI Publisher Server that contains the schedule job information.

This task is a step in ["Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations"](#) on page 33.

To add a container URL for Oracle BI Publisher Server

- 1 Navigate to the Administration - Server Configuration screen, Enterprises, and then the Profile Configuration view.
- 2 Select the XMLPJvmSubsys profile.
- 3 In the Profile Parameters subview, set the value of the Container URL field as follows:

`http://<serverName>:<Port>/siebel/jbs`

where:

Port is the port number used for the config agent port parameter.

- 4 Restart the Siebel Services.

Implementing EAI File Handling Changes for Siebel Reports Generation

For Siebel Fix Packs 8.1.1.11 and 8.2.2.4, a new `EAIFileTransportFolders` enterprise parameter was introduced. This parameter allows you to restrict write access for the EAI File Transport to specific folders within the Siebel file system. However, for report generation to work properly, you must change the value of this parameter specific to your deployment. For more information about this new feature, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

This task is a step in ["Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations"](#) on page 33.

NOTE: This task applies to both Windows and UNIX environments.

CAUTION: For interactive and scheduled report generation to work properly, you must set the `EAIFileTransportFolders` enterprise parameter, otherwise report generation fails.

To implement EAI file handling changes for Siebel Reports generation

- 1 Navigate to the Administration - Server Configuration screen, Enterprises, and then the Parameter view.
- 2 In the Name field in the Enterprise Parameters list, query for EAI File Transport Folder List.
- 3 For the value, enter the following:
`c: \SIEBSRVR_ROOT\ses\si_ebsrvr\xml p\reports; c: \SIEBSRVR_ROOT\ses\si_ebsrvr\xml p\data`

where:

`SIEBSRVR_ROOT` is the actual path where the Siebel Server is installed.

NOTE: For report scheduling, you must first add the `SIEBSRVR_ROOT\ses\si_ebsrvr\xml p\data` directory. By default, this directory is not available.

- 4 Restart the Siebel Server.
- 5 (For Siebel Mobile Web Client and Siebel Developer Client only), do the following:
 - a Add a new section in the `application.cfg` file as follows:

```
[EAIFileTransportConfigSubsys]
EAIFileTransportFolders = c: \SIEBSRVR_ROOT\ses\si_ebsrvr\xml p\reports;
c: \SIEBSRVR_ROOT\ses\si_ebsrvr\xml p\data
```
 - b Update the following section in the `application.cfg` file as follows:

[XMLPJvmSubsys]

Full Name = XMLPJvmSubsys

Description = XMLP Java Business Service Subsystem Parameters

SubsysType = JavaContainerSubSys

CONTAINERURL = http://<ServerName>:<port>/siebel/jbs

where:

Port is the port number used for the config agent port parameter.

For more information, or if you receive error message SBL-RPT-50524 or SBL-EAI-50228 (even if you have applied Siebel CRM Fix Pack 8.1.1.11), see 1551684.1 and 1542583.1 (Article IDs) on My Oracle Support.

Uploading Preconfigured Report Layout Templates to Oracle BI Publisher Server

Siebel CRM ships with preconfigured report layout templates. To make these report layout templates accessible to users in connected mode, you must upload them to Oracle BI Publisher Server. For a comprehensive list of the preconfigured reports that ship with Siebel CRM, see 876284.1 (Article ID) on My Oracle Support.

NOTE: The instructions in this topic are not applicable for new reports you create or modify while in disconnected mode. Such reports require deployment to the Siebel Web Client.

CAUTION: If the report layout templates are not uploaded to Oracle BI Publisher Server, when a user attempts to run a report, an error occurs and the report fails.

This task is a step in ["Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations"](#) on page 33.

To upload preconfigured report layout templates to Oracle BI Publisher Server

1 Navigate to the Administration - BI Publisher Reports screen, then the Reports - Standard Templates view.

2 From the application-level menu, select Edit, and then Select All.

NOTE: If you receive a SBL-RPT-50582 error message, upload the preconfigured reports individually. This error means that a valid XLIFF or ZIP file has not been associated with the report.

3 Click Upload Files.

NOTE: When you create your own custom reports, you must upload the report template and associate it with a view before you can use that report.

- 4 Navigate to the Administration - BI Publisher Reports screen, then the View Association view to associate the Account List report to the Account List view.

The Account List report layout template is now available to the Siebel application for use in generating reports.

- 5 Verify Oracle BI Publisher integration by logging in to Oracle BI Publisher Server with administrative privilege.
- 6 Test report generation by performing the following:
 - a Navigate to the Accounts screen, then the Accounts List view.
 - b Perform a query that returns approximately 30 records.
 - c In the application toolbar, click the Reports button.
 - d In the BI Publisher section, choose Account List.
 - e (Optional) Select the report output type, and then click Submit.

NOTE: If you receive error message SBL-RPT-50524 or SBL-EAI-50228, even though you have applied Siebel CRM Fix Pack 8.1.1.11 (or 8.2.2.4), see 1542583.1 (Article ID) on My Oracle Support for resolution.

Process of Enabling and Configuring Report Scheduling

Before you can use report scheduling, you must perform several configuration tasks for both the Siebel application and Oracle BI Publisher.

This process is a step in [“Roadmap for Integrating Oracle BI Publisher with Siebel CRM for First-Time Installations”](#) on page 33.

To enable and configure report scheduling, perform the following tasks:

- 1 [“Configuring the Data Service WSDL for Oracle BI Publisher Server”](#) on page 57
- 2 [“Verifying Symbolic URL Arguments That Embed the Oracle BI Publisher Scheduling Views in the Siebel Application”](#) on page 58

Configuring the Data Service WSDL for Oracle BI Publisher Server

Oracle BI Publisher Server uses a Web service to pull data from the Siebel application when a report is scheduled and executed. The WSDL (Web Service Definition Language) provides a definition of this Web service. Oracle BI Publisher Server must know where this WSDL file is located.

This task is a step in [“Process of Enabling and Configuring Report Scheduling”](#) on page 57.

To configure the data service WSDL for Oracle BI Publisher Server

- 1 Log in to the Siebel application as a Siebel administrator.
- 2 Navigate to the Administration - Web Services screen, then the Inbound Web Services view.
- 3 In the Name field in the Inbound Web Services list, query for BIPDataService.
- 4 Change the host, port, and locale of the server port's address to that of the Siebel Server as following:

```
https://<Server>:<Port>/siebel/app/eai_anon/  
<Lang>?SWEEExtSource=SecureWebService&SWEEExtCmd=Execute&WSSOAP=1
```

where:

- *Server* is the same as the server name on which the Web service resides.
- *Lang* is the three-letter designator for the locale of the Siebel Server (for example, enu).

NOTE: Use the same URL for both SOAP header-based authentication and URL-based authentication.

- 5 Select BIPDataService, and then click Generate WSDL.
- 6 Save the WSDL as a new data service file named `dataservice.wsdl` in a directory in the Oracle BI Publisher home directory (by default, this is the `<Installation>\user_projects\domains\bi foundation_domain` directory).

For example, you might save the file as:

```
<Installation>\user_projects\domains\bi foundation_domain\dataservice.wsdl
```

NOTE: You must use lower case characters to match existing references.

The WSDL is now available for importing.

Verifying Symbolic URL Arguments That Embed the Oracle BI Publisher Scheduling Views in the Siebel Application

The Scheduled Reports view in the Siebel application is for managing scheduled reports. This view is an embedded Web page from the Oracle BI Publisher Enterprise application. To configure this view, you implement a symbolic URL to display external contents in the Siebel user interface by specifying how to construct the external application and define arguments and values that are passed to the Siebel client. For more information about working with symbolic URLs, see *Siebel Portal Framework Guide*.

This task is a step in [“Process of Enabling and Configuring Report Scheduling”](#) on page 57.

To verify symbolic URL arguments that embed the Oracle BI Publisher Scheduling Views in the Siebel application

- 1 Navigate to the Administration - Integration screen, WI Symbolic URL List, and then the Host Administration view.
- 2 Query for biphost virtual name, and then edit for your deployment.
- 3 Navigate to the Symbolic URL Administration view.
- 4 Query for BIBReportJobListPage, and then set the following values:
 - URL is *http://biphost/xmlpserver/servlet/viewHistory*
 - Name is *host:port*
 - Host Name is *host:port*
 - Fixup Name is Default
 - SSO Disposition is IFrame

where *host:port* is the Oracle BI Publisher host address you choose from a picklist and the URL value is added as part of the seed data.

- 5 Add new records in Symbolic URL Arguments as shown in the following table.

Argument Name	Required Argument	Argument Type	Argument Value
PreloadUrlPerSession	Yes	Command	True
uid	Yes	Field	BIPUser
anyname	Yes	Command	PostRequest
passwd	Yes	Field	BIPPassword
_xuil	Yes	Field	BIPLanguage
NOTE: Applies for non-English deployments only.			
pgheader	Yes	Constant	hide

- 6 Restart the Siebel Server.

About Upgrading Siebel Reports for Siebel Innovation Pack 2017

If you are currently using Siebel Reports and you plan to upgrade to Siebel Innovation Pack 2017, you must do the following:

- Upgrade your Siebel application (or applications).

You must have performed the Siebel incremental repository merge before you can use the latest Siebel Reports features. For information on performing an incremental repository merge, see *Siebel Database Upgrade Guide*. For an overview of enhancements that can be implemented using incremental repository merge, as well as different installation options for the fix pack, see *Siebel Maintenance Release Guide* on My Oracle Support.

- Upgrade to the latest Oracle BI Publisher patch.

For information on upgrading to the latest Oracle BI Publisher patch, see [“Upgrading to the Latest Oracle BI Publisher Patch” on page 41](#).

5

Administering Siebel Reports

This chapter describes some of the administrative tasks for Siebel Reports. It includes the following topics:

- [Doing Typical Administrative Tasks on page 61](#)
- [Optimizing the Performance of Siebel Reports on page 65](#)
- [Administering Multilingual Reports on page 71](#)
- [Administering Signatures in Reports on page 75](#)

Doing Typical Administrative Tasks

This topic describes how to do typical administrative tasks. It includes the following information:

- [Disabling the Reports Button on page 61](#)
- [Making Sure Siebel CRM Displays The Report Status on page 62](#)
- [Making Sure Fonts Are Available for Reports on page 62](#)
- [Managing Report Notifications on page 63](#)
- [Viewing Report Usage Statistics on page 63](#)
- [Configuring Siebel CRM to Automatically Delete Reports on page 63](#)

Disabling the Reports Button

Siebel CRM displays the Reports button in the Siebel client in the Application toolbar, by default. You can disable or enable it. If you disable it, then you can still use a Siebel workflow process or business service to run reports. For more information, see [“About the Report Business Service” on page 83](#).

To disable the Reports button

- 1 Log in to the Siebel application with administrator privileges.
- 2 Navigate to the Administration - Application screen, and then the System Preferences view.
- 3 Query the System Preference field for ReportEngineType, and then set the System Preferences value using values from the following table.

System Preference Value	Description
NONE	Configures Siebel CRM to not display the Reports button.
BIP	Configures Siebel CRM to display the Reports button.

- 4 Restart the Siebel Server, and then verify the state of the Reports button.

Making Sure Siebel CRM Displays The Report Status

Siebel CRM uses the submit time on the Siebel Server to determine the timestamp that it uses for a report. If a user runs a report, and if the XMLP Report server component is not enabled, then Siebel CRM displays an empty status for the report in the My Reports view. If you subsequently enable the XMLP Report server component, then Siebel CRM immediately displays a status of In Progress. However, it might not display the status long enough for you to view it. This situation occurs for most reports because Siebel CRM displays the In Progress status only for a few seconds, and then replaces it with a status of Success or Error.

To make sure Siebel CRM displays the report status

- Enable the XMLP Report server component.

Making Sure Fonts Are Available for Reports

Siebel CRM uses the fonts that are available on the computer where it runs the report. If the fonts that a report requires are not available on this computer, then the report output might be different than what you expect. This topic describes how to copy fonts on the Siebel Server so that they are available in the report output. For more information about using fonts in Siebel CRM, see *Siebel Global Deployment Guide*.

To make sure fonts are available for reports

- 1 Log on to the Siebel Server where you enabled the XMLP Report server component.
- 2 Copy the fonts, depending on the following operating system that your deployment uses:
 - **Windows.** Copy the fonts that the following folder contains to the fonts subfolder that resides in the Java installation folder for your environment. This subfolder is typically `JAVA_HOME\lib\fonts`:
 - `C:\WINDOWS\Fonts`
 - **UNIX.** Copy the fonts that reside in the `/Fonts` folder to the fonts subfolder in the Java installation folder for your environment. For example:
 - **Solaris.** `/usr/openwin/lib/X11/fonts/TrueType`
 - **AIX.** `/usr/lpp/X11/lib/X11/fonts/TrueType`
 - **HP.** `/usr/lib/X11/fonts/ms.st/typefaces`
 - **Linux.** `/usr/lib/X11/fonts/TrueType`

Managing Report Notifications

Siebel CRM enables report notification in the Siebel Open UI client, by default. You can disable or enable it. If you enable report notification, then Siebel CRM does the following:

- If the user clicks Submit, then Siebel CRM immediately returns to the Siebel application. The user is not required to wait for the report to finish running. The user can navigate to another view or do some other work while the report runs in the background.
- Displays a notification in the report icon when the report finishes running.
- Allows the user to view report notifications and to open a completed report without navigating to the My Reports view.
- Hides the notification when the user opens the report.
- Displays a report notification only to the user who runs the report, and only for the current user session. It does not display the notification to any other user. It does not display the notification if the user logs out and then logs back in to the client.

To disable reports notification

- 1 Log in to the Siebel application with administrator privileges.
- 2 Navigate to the Administration - Application screen, and then the System Preferences view.
- 3 Query the System Preference field for Report Notification Mode.
- 4 Set the System Preference Value to Off.
- 5 Restart the Siebel Server.

Viewing Report Usage Statistics

You can use the predefined Report Usage Statistics report to get detailed information about reports, the users who run reports, failures, long-running reports, and so on. For information about downloading a predefined report, see 876284.1 (Article ID) on My Oracle Support.

Configuring Siebel CRM to Automatically Delete Reports

This topic describes how to configure Siebel CRM to automatically delete reports from the Siebel File System and the Oracle BI Publisher Server which frees up memory.

CAUTION: To avoid deleting or corrupting the report repository, it is recommended that only an administrator configure Siebel CRM to automatically remove reports from the Siebel File System or the Oracle BI Publisher Server. Business users and other users can delete individual report instances, but they must configure Siebel CRM to automatically delete all reports. For more information, see ["Deleting Reports" on page 24](#) and ["Types of Users Who Use Siebel Reports" on page 11](#).

Deleting Reports from the Siebel File System

It is recommended that you configure Siebel CRM to regularly delete reports from the Siebel File System.

To delete reports from the Siebel File System

- 1 Log in to the Siebel application with administrator privileges.
- 2 Navigate to the Administration - Application screen, and then the System Preferences view.
- 3 In the System Preferences list, query the System Preference Name field for BIP Delete After Days.
- 4 Modify the value in the System Preference Value field to a positive, nonzero value.

Siebel CRM sets the value to -1 (minus 1), by default. This preference determines how frequently Siebel CRM runs the Auto Purge workflow process. This workflow removes reports from the database and the Siebel File System after the number of days that you specify. For example, to keep only the reports that Siebel CRM created during the most recent seven days, you set the value to 7. Siebel CRM then deletes all reports that are more than seven days old.

It is recommended that you set this value to the lowest value possible while still meeting your report requirements, particularly in a large deployment where users create many reports. Setting this parameter to a low value prevents the Siebel File System from increasing to an unmanageable size.

For more information about modifying the Auto Purge workflow process, or using a workflow process to schedule a job to run periodically, see *Siebel Business Process Framework: Workflow Guide*.

- 5 Navigate to the Administration - Server Management screen, and then the Jobs view.
- 6 Add a new job named Workflow Process Manager.
- 7 In the Job Parameters list, add a parameter to the job you added in [Step 6](#). Use values from the following table:

Name	Value
Workflow Process Name	XMLP Purge Records

- 8 Click Submit.
- 9 Restart the Siebel Server.

Deleting Scheduled Reports from the Oracle BI Publisher Server

Siebel CRM stores scheduled reports that users create in the Oracle BI Publisher scheduling database. It is recommended that you configure Siebel CRM to regularly delete these reports. You cannot remove scheduled reports from the Siebel application or by using the Oracle BI Publisher administration application.

To remove scheduled reports from the Oracle BI Publisher Server

- Use the Job Manager Instances List or the Job Manager Purge Instances Window to remove Oracle Business Intelligence Scheduler job instances from the scheduling database.

For more information, see the topic about Purging Oracle BI Scheduler Job Instances in the *Oracle Business Intelligence Scheduler Guide* on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

Optimizing the Performance of Siebel Reports

This topic describes how to optimize the performance of Siebel Reports. It includes the following information:

- [“Setting the Waiting Period” on page 65](#)
- [“Setting the Polling Interval” on page 66](#)
- [“Setting Concurrency Parameters” on page 66](#)
- [“Optimizing Reports That Include Thousands of Records” on page 67](#)

It is recommended that you do this work when you install Siebel Reports. However, you can do this work at any time. You must use a connected client to do this work, unless noted otherwise.

For more information about improving performance with:

- Integration objects that you customize, see [“Reducing the Amount of Data That Integration Objects Transfer” on page 104](#)
- Running reports, see 1392449.1 (Article ID), Siebel BI Publisher Report Generation, on My Oracle Support
- Integrating reports, see 1466709.1 (Article ID), Siebel CRM BI Publisher Integration Performance, on My Oracle Support

Setting the Waiting Period

The waiting period sets the time that Siebel CRM waits before it sends a report to run in the background. It is recommended that you set this time. For example, if a report requires 90 seconds to finish running, and if you set the wait time to 60, then Siebel CRM does the following:

- Runs the report in the foreground for 60 seconds. The user cannot use the client during these 60 seconds.
- Displays a message after 60 seconds that is similar to the following:

XMLP Report Generation time exceeds the Threshold time. Please see the generated Report Output File in the BIP my Reports view.

The user can start using the client immediately after dismissing this message.

- Adds the finished report to the My Reports view after 90 seconds.

To set the waiting period

- 1 Log in to the Siebel application with administrator privileges.
- 2 Navigate to the Administration - Application screen, and then the System Preferences view.
- 3 In the System Preferences list, choose BIP Report Wait Time, and then modify the System Preference Value.

The minimum value is 0, which configures Siebel CRM to send all reports to the background immediately. For information about this view, see [“Running Reports in Siebel CRM Clients” on page 19](#).

- 4 Restart the Siebel Server.

Setting the Polling Interval

It is recommended that you change the Database Polling Interval that the Server Request Processor uses.

To set the polling interval

- 1 Log in to the Siebel application with administrator privileges.
- 2 Navigate to the Administration - Server Configuration screen, Servers, and then the Components view.
- 3 In the Components list, choose Server Request Processor.
- 4 Scroll down, click the child Parameters list, and then click Hidden.
- 5 In the Parameter list, choose Database Polling Interval, and then change the value from 10 to 1.

This parameter determines how long Siebel CRM waits before the Server Request Processor polls the database to determine if the server queue includes a new report request. Siebel CRM also updates the Value on Restart and Default Values.

- 6 Restart the Siebel Server.

You can also use the `srvrmgr` command-line interface to restart the Server Request Processor.

For more information about setting the server component parameters, and restarting the Siebel Server and server components, see *Siebel System Administration Guide*.

Setting Concurrency Parameters

It is recommended that you set the concurrency parameters.

To set concurrency parameters

- 1 Log in to the Siebel application with administrator privileges.

Log in on the computer where the XMLP Report server component is enabled. If you prefer, you can use the Server Manager instead of the administrative screens to set these parameters. For more information, see [“Using the Server Manager to Set Concurrency Parameters” on page 67](#).

- 2 Navigate to the Administration - Server Configuration screen, Servers, and then the Components view.
- 3 In the Components list, choose XMLP Report.
- 4 Click the Parameters tab, and then set the parameters using values from the following table.

Parameter	Description
Maximum Tasks parameter	Modify the Value on Restart to 100.
Maximum MT Servers	Modify the Value on Restart to 2.

Using the Server Manager to Set Concurrency Parameters

Use the following procedure to set the concurrency parameters using the Server Manager (srvmgr).

To use the Server Manager to set concurrency parameters

- Run the following Server Manager (srvmgr) commands to increase the MaxTasks parameter to 100 and the MaxMTServers parameter to 2:

```
change param maxtasks=100 for comp xmlpreportserver server Siebel Server
change param maxmtservers=2 for comp xmlpreportserver server Siebel Server
```

where:

- *SiebelServer* identifies the name of the Siebel Server.

Optimizing Reports That Include Thousands of Records

This topic describes how you can optimize performance for reports that include a lot of records, such as 10,000 records.

CAUTION: It is recommended that you do the work that this topic describes on a separate, named subsystem that only the XMLP Report server component uses. For example, reportsdatascr. If you do not do this, then the Application Object Manager (AOM) on the client might fail if it gets a large amount of data.

If a report includes a lot of records, then it is recommended that the user schedule this report to run on the Oracle BI Publisher Server. It is recommended that the user run this report immediately. For more information, see [“Scheduling Reports” on page 25](#).

Modifying the HTTP Sleep Time

The integration from Siebel CRM to Oracle BI Publisher uses an outbound Web service call that calls the PublicReportServiceService Web service on the Oracle BI Publisher Server. The outbound Web service uses the EAI HTTP Transport business service to send the HTTP request for this call. Siebel CRM sets the HTTPSleepTime argument of this business service to two minutes (120000 milliseconds), by default. If a report requires over two minutes to finish, and if you do not schedule this report, then the report times out and Siebel CRM adds an entry that is similar to the following error message. It adds this message in the XMLPReportServer .log file:

```
SBL-EAI -04311: Operati on ' runReport' is expecting a response but no response was
recei ved.
```

This error typically occurs only with a large or complex report, and it does not occur if you schedule the report. To avoid this error, you can increase the default sleep time for the EAI HTTP Transport business service so that the EAI component keeps the HTTP requests that are open through the Outbound Web Service long enough so that Siebel CRM can finish running the report.

To modify the sleep time for all reports

- 1 Log in to Siebel client with administrative privileges.
- 2 Navigate to the Administration - Server Configuration screen.
- 3 Query the Component field of the Components list for XMLP Report Server.
- 4 In the Components list, click the child Parameters tab.
- 5 In the Component Parameters list, query the Parameter field for the following value:

ReportHTTPSleepTime

- 6 In the Value field, set the value to the number of milliseconds.
For example, set the Value to 240000 to specify a four minute timeout.

NOTE: The default, and maximum, value for the ReportHTTPSleepTime parameter is 180 seconds.

- 7 Restart the server.

Using the Business Service Simulator to Test the Sleep Time for a Single Report

You can use the Business Service Simulator to test the timeout for a single report without affecting other reports. You can use it to do a variety of tasks, such as to run a workflow process that creates a report. You can test the sleep time for a single report only through the XMLP Driver Service business service. For more information about using a business service, see *Integration Platform Technologies: Siebel Enterprise Application Integration*, and *Configuring Siebel Business Applications*.

To use the Business Service Simulator to test the sleep time for a single report

- 1 Log in to Siebel client with administrative privileges.
- 2 Navigate to the Administration - Business Service screen, and then the Simulator view.

- 3 In the Simulator list, click New, and then set the fields using values from the following table.

Field	Value
Service Name	XMLP Driver Service
Method Name	GenerateBIPReport
Iterations	1

- 4 In the Input Arguments List, click New, and then add the HTTPSleepTime input argument. The HTTPSleepTime input argument affects only the report that you specify in the ReportName input argument in [Step 5](#). For example, if you set HTTPSleepTime to 120000, and if you set ReportName to Opportunity List, then Siebel CRM times out after two minutes only for a single instance of an opportunity report.
- 5 Add all other input arguments that the business service simulator requires. The XMLP Driver Service includes a number of other required input arguments that you must add. For information about these input arguments and how to add them, see 1425724.1 (Article ID), Siebel BI Publisher Reports Business Service Methods, on My Oracle Support.

Setting Server Parameters

This topic describes how to set server parameters.

This task is a step in [“Optimizing Reports That Include Thousands of Records”](#) on page 67 and is applicable with a connected client.

To set server profile parameters

- Set the DSMaxFetchArraySize server parameter to -1 (negative one).

Enabling Scalable Mode

This topic describes how to set the scalable mode for all reports. Setting the scalable mode prepares the BI Publisher-FO Processor to handle a large amount of data, longer processing times, time-outs, and so on. Scalable mode is slower than nonscalable mode but it makes sure data loss does not occur even with a large, complex report.

It is recommended that you enable scalable mode only if many users must access the same, large report at the same time. Scalable mode consumes more resources and might degrade performance. If your deployment commonly uses large reports, then it is recommended that you set the scalable mode for all reports.

For information about the BI Publisher-FO Processor, see [“Overview of How Siebel CRM Runs Reports in Clients”](#) on page 78.

To enable scalable mode

- 1 Log on to the Oracle BI Publisher Server.

- 2 Navigate to the following folder:

```
\BI\user_projects\domains\bi_foundation_domain\config\bi_publisher\repository\Admin\Configuration
```

The path for the Java installation folder varies depending on where you install JRE.

- 3 Use a text editor to open the xdo.cfg file.
- 4 Locate the properties tag.
- 5 Make sure the Scalable Mode parameter is set to true:

```
<property name="xslt-scalable">true</property>
```

- 6 Save the xdo.cfg file.

Configuring a Temporary Folder

You can configure a temporary folder that Siebel CRM can use if you run a large number of reports.

To configure a temporary folder

- 1 Do [Step 1 on page 69](#) through [Step 4 on page 70](#).
- 2 Configure a temporary folder on the Oracle BI Publisher Server. Use the following code:

```
<property name="system-temp-dir">path</property>
```

where:

- *path* is the path to the temporary folder

For example:

```
<property name="system-temp-dir">d:\tmp</property>
```

This temporary folder provides the Oracle BI Publisher Server with the disk space it requires to create complex reports. You must make sure the location of the temporary folder possesses adequate space for the temporary files. You can delete files from this folder after the reports finish running.

Save the xdo.cfg file.

Increasing Memory for the Java Virtual Machine

The *Java virtual machine heap* is the area of memory that the Java Virtual Memory uses for dynamic memory. If users must run a report that includes a large data set, then the WebLogic process might consume too much Java Virtual Memory. To avoid a reports failure, you can increase the allocation that the Java Virtual Memory uses for the WebLogic process.

To increase memory for the Java Virtual Machine

- 1 Modify the Java Virtual Memory for the Administration Server by specifying the following value in the setDomainEnv.sh script:

```
----USER_MEM_ARGS="-Xms2048m -Xmx2048m" export USER_MEM_ARGS----
```

- 2 Modify the Java Virtual Memory for the Siebel Management Server, by specifying the following value in the setDomainEnv.sh script:

```
----MEM_ARGS="-Xms4096m -Xmx4096m" export MEM_ARGS
```

For more information, see see 1272019.1 (Article ID) on My Oracle Support.

Administering Multilingual Reports

This topic describes how to administer reports that use more than one language. It includes the following information:

- [“How Siebel CRM Runs Multilingual Reports” on page 71](#)
- [“Creating Multilingual Reports” on page 73](#)

How Siebel CRM Runs Multilingual Reports

Siebel CRM uses a single object manager to run reports that include multiple languages. It does the following work:

- Allows you to configure reports for a specific locale.
- Makes sure a report runs for a record that includes a locale preference. For example, to make sure Siebel CRM creates a Quote report according to the language preference that the quote record specifies.
- Allows the user to set a default locale preference for the reports that this user runs.
- Allows the user to override the default locale values when running reports.
- Uses the default locale settings of the object manager that Siebel CRM uses for the user session.

Typically, you manage how Siebel CRM does report translations. A third-party typically does the actual translation.

For more information about:

- Languages that Siebel CRM supports, see *Siebel System Requirements and Supported Platforms* on Oracle Technology Network.

NOTE: For Siebel CRM product releases 8.1.1.9 and later and for 8.2.2.2 and later, the system requirements and supported platform certifications are available from the Certification tab on My Oracle Support. For information about the Certification application, see article 1492194.1 (Article ID) on My Oracle Support.

- Installing Siebel Language Packs, see *Siebel Installation Guide* for the operating system you are using.
- Deploying multiple languages, see *Siebel Global Deployment Guide*.

How Siebel CRM Translates User Interface Elements

To translate a report to another language, such as from English to German, Siebel CRM exports user interface elements according to the multiple languages that your deployment requires. It exports the strings it uses in a report to an XLIFF file. The XLIFF format separates localizable text from formatting, which allows the translator to work only on the text that it must translate. Siebel CRM does the following work for each report locale when it creates a multilingual report:

- **Determines template text.** Uses the translations that the XLIFF file specifies.
- **Translates MLOVs.** Uses the report locale and LOV data that it gets from EAI for each language.
- **Formats date, time, and currency.** Uses the following template functions:
 - format-date
 - format-currency

These Oracle BI Publisher functions use the locale code that you specify for the report template.

For more information about how Siebel CRM uses XLIFF files, see [“Directory Structure That Siebel Reports Uses”](#) on page 86.

Priority That Determines the Report Language

[Table 5](#) describes the priority that Siebel CRM uses to determine the report language it uses when it runs a report.

Table 5. How Siebel CRM Determines the Language It Uses When It Runs a Report

Priority	Locale Preference	Description
1	User chooses	If the user chooses a language in the Run Report pane, then Siebel CRM uses this language when it runs the report. For more information, see Chapter 3, “Running Reports.”
2	Report template	If priority 1 does not specify a locale, then Siebel CRM uses the default locale that the report template specifies. For more information, see “Creating Multilingual Reports” on page 73.
3	Business service	If priority 1, and 2 do not specify a locale, then Siebel CRM uses the locale that the Report Business Service specifies. If Siebel CRM sends no parameters to the Report Business Service, then it uses the default locale that the application object manager specifies. For more information, see “About the Report Business Service” on page 83.

Table 5. How Siebel CRM Determines the Language It Uses When It Runs a Report

Priority	Locale Preference	Description
4	User preference	If priority 1, 2, and 3 do not specify a locale, then Siebel CRM uses the user preference that the user sets in the client. For information about setting user preferences, see “Setting User Preferences for Siebel Reports” on page 19.
5	No language or locale specified	If priority 1, 2, 3, and 4 do not specify a locale, then Siebel CRM uses the locale that the application object manager uses. It uses the application object manager that is running for the current user session.

Creating Multilingual Reports

You create a multilingual report in the same way that you create a report that is not multilingual. The only difference is that you must externalize the user interface elements so that they support multiple languages in the locales that you require. For more information about externalizing the user interface elements, see [“How Siebel CRM Translates User Interface Elements” on page 72.](#) For more information about designing reports, see [Chapter 7, “Customizing Siebel CRM Reports.”](#)

If you add a locale to a report template, then Siebel CRM creates a separate report for each locale that you add. It creates these reports in addition to the report it creates according to the priority described in [“Priority That Determines the Report Language” on page 72.](#)

To create a multilingual report

- 1 Navigate to the Administration - Data screen, and then the List of Values view.
- 2 Query for XMLP_RPT_OUTPUT_TYPE and set the Display Value setting to ALL.
- 3 Select the following check boxes:
 - Active
 - Translate
 - Multilingual
- 4 (Optional) Repeat [Step 2](#) and [Step 3](#) for each output type you require.
- 5 Design a report in the default language.
ENU (English U.S.) is an example of a default language.
- 6 Use Oracle BI Publisher Desktop to redesign the report template so that it can accommodate the spatial layout of the primary language that your deployment uses.
For more information, see [“About Oracle BI Publisher Desktop” on page 15.](#)
- 7 (Optional) Use BI Publisher Desktop to add a default locale to the report template:
 - a Use Oracle BI Publisher Desktop to open the report template.
 - b In the Oracle BI Publisher menu, choose Tools, Translate Template, and then Localize Template.

- Do the work described in “[Registering Report Templates](#)” on page 93, with the following differences:

- Use the following format when you name the XLIFF or RTF file that you upload:

TemplateName_language code_TERRITORY CODE.extension

where:

- *TemplateName* is the name of the report template.
- *language code* is a lower-case, two-letter ISO language code.
- *TERRITORY CODE* is an upper-case, two-letter ISO country code.
- *extension* is xlf or rtf.

For example, if the template name is EmployeeTemplate, and if you must upload a Japanese-Japan translation, then use the following file name:

EmployeeTemplate_ja_JP.xlf

- Use the following format when you name the ZIP file that you upload:

report template file name.zip

Files names are not case-sensitive.

- In the child Translations list, click New to add a new language.
 - In the Language field, add a new language, and then click OK.
 - (Optional) Enter a Report Name.

Siebel CRM displays the name you enter for the translated report in the Run Report pane.

CAUTION: Each translated report must include a unique, translated display name. If duplicate names for a translated report exists, then the Siebel Server cannot determine the report to run when Siebel CRM sends these names from the browser to the Siebel Server, and the report might fail.

When you click Upload, Siebel CRM does the following work according to the type of file you upload:

- **RTF file.** Stores the RTF file in the *SIEBEL_ROOT\XMLP\TEMPLATES* folder, and then uploads the RTF and XLIFF files a language subfolder on the Oracle BI Publisher Server.
- **ZIP file.** Stores the ZIP file in the *SIEBEL_ROOT\XMLP\xliff* folder, and then decompresses this file to the following folder:

SIEBEL_ROOT\XMLP\xliff\language_code

For more information about this directory structure, see “[Directory Structure That Siebel Reports Uses](#)” on page 86.

Siebel CRM validates these files during the upload. If it finds an error, then it displays an error message and deletes the uploaded files. For information about this validation, see “[Validations That Siebel CRM Does When You Upload a Report Template](#)” on page 96.

- When you validate the directory structure of the Oracle BI Publisher Server, note that Siebel CRM uses the following format to register each XLIFF file:

TemplateName_language code_TERRITORY CODE.xlf

For more information about using this format, see *Oracle Business Intelligence Publisher Report Designer's Guide* on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

Administering Signatures in Reports

This topic describes how to administer reports that contain signatures. It includes the following information:

- [“Creating and Modifying an Integration Object for Signatures in Reports” on page 75](#)
- [“Customizing a Report Template for Signatures” on page 76](#)

Before You Begin. You must ensure that the SiebelCustomXMLP_SIA.jar Siebel Java archive (JAR) file is included in the class path for XMLPJvmSubsys along with InkToolsLib.jar and iSignBmp.jar. For example:

```
C:\classes\SiebelXMLP.jar; C:\classes\Siebel.jar; C:\classes\XSLFunctions.jar; C:\classes\SiebelCustomXMLP.jar; C:\classes\SiebelCustomXMLP_SIA.jar; C:\classes\InkToolsLib.jar; C:\classes\iSignBmp.jar
```

For more information, see [“Copying JAR Files to Oracle BI Publisher Server” on page 39](#).

Creating and Modifying an Integration Object for Signatures in Reports

This topic describes how to create and modify an integration object so that you can capture signatures in your reports.

To create and modify an integration object for signatures in reports

- 1 Create a new integration object.
For more information, see [Creating New Integration Objects to Add Fields to Reports on page 102](#).
- 2 In the Object Explorer, expand the Integration Object tree, expand the Integration Component tree, and then click Integration Component Field.
- 3 Scroll through the Integration Component Field list until you locate the Signature field.
- 4 Confirm the following Signature field properties:
 - **XML Tag.** ssSignature

- ID. sslid

NOTE: If you use signatures migrated from other applications in BI Publisher, then you must ensure that the field for migrated signature data has the value `ssSignatureImg`.

Customizing a Report Template for Signatures

This topic describes how to modify an existing report template to capture signatures in reports.

NOTE: For more information on creating report templates, see [“Creating Report Templates” on page 91](#).

To customize a report template for signatures

- 1 Open Oracle BI Publisher Desktop.
For more information, see [“About Oracle BI Publisher Desktop” on page 15](#).
- 2 Select and open the OOTB LS_SIGNAD.rtf template in the following folder:
SIEBEL_SERVER_ROOT\XMLP\TEMPLATES
- 3 In the OOTB LS_SIGNAD.rtf template, double-click the fields and rename them as necessary.
For example, if you want to use this report to obtain signatures for an invoice, then you can replace `<?for-each:ssPharmaCallSignature?>` with `<?for-each:ssFsInvoiceSignature?>`.
- 4 In the OOTB LS_SIGNAD.rtf template, remove the following fields:
 - `ssContactLast`
 - `ssContactFir`
 - `concat(concat(ss`
- 5 Save the report template in RTF format.

6

Siebel Reports Architecture

This chapter describes the Siebel Reports architecture. It includes the following topics:

- [About the Siebel Reports Architecture on page 77](#)
- [How Siebel CRM Controls Access to Reports on page 87](#)

About the Siebel Reports Architecture

This topic describes the architecture that Siebel Reports uses. It includes the following information:

- [“Overview of How Siebel CRM Runs Reports in Clients” on page 78](#)
- [“How Siebel CRM Runs Reports in Connected Clients” on page 79](#)
- [“How Siebel CRM Runs Reports in Disconnected Clients” on page 81](#)
- [“How Siebel Reports Uses Integration Objects” on page 82](#)
- [“Computer Languages That Siebel Reports Uses” on page 83](#)
- [“About the Report Business Service” on page 83](#)
- [“Directory Structure That Siebel Reports Uses” on page 86](#)

Siebel CRM uses the following separate architectures starting with Siebel CRM versions 8.1.1.x and 8.2.2.x:

- **Connected architecture.** Siebel CRM integrates directly with the Siebel XMLP Report server component and the Oracle BI Publisher Server.
- **Disconnected architecture.** Siebel CRM uses Oracle BI Publisher to run Oracle BI Publisher through the XMLP Report Business Service.

Overview of How Siebel CRM Runs Reports in Clients

Figure 5 illustrates an overview of how Siebel CRM runs reports in connected and disconnected clients.

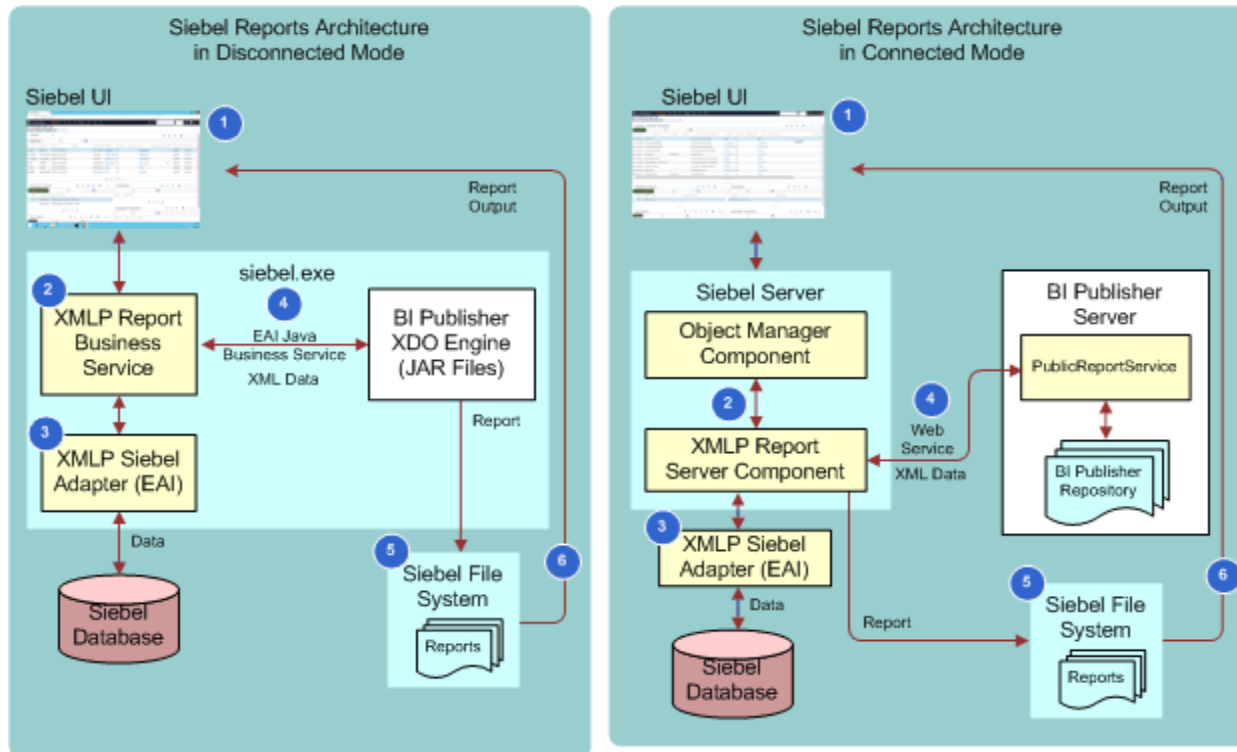


Figure 5. Overview of How Siebel CRM Runs Reports in Connected and Disconnected Clients

Explanation of Callouts

Siebel CRM does the following work when it runs a report in a connected or disconnected client:

- The user requests to run a report in the Siebel client, and then Siebel CRM sends this request to one of the following items:
 - Oracle BI Publisher Server if the user is using a connected client
 - Oracle BI Publisher XDO Engine if the user is using a disconnected client
- The RTF Processor converts the report template files to the XSL format so that Siebel CRM can use it as input to the BI Publisher-FO Processor.
- The BI Publisher-FO Processor merges the XSL and the XML data files, and then creates one or more of the following output formats, depending on the output type that the user chooses:
 - Converts XML and XSL formats to HTML.

- Converts XML, XSL-FO, and XLIFF formats to PDF, HTML, RTF, EXCEL, PPT, and MHTML.

For more information about the file types that the user can choose, see [“Output File Types That Siebel Reports Supports”](#) on page 14.

Oracle renamed XML Publisher and XMLP to Oracle BI Publisher. Some items still use XML Publisher or XMLP.

How Siebel CRM Runs Reports in Connected Clients

Figure 6 illustrates the architecture that Siebel CRM uses to run a report if the user is using a Siebel client that is connected to a Siebel Server.

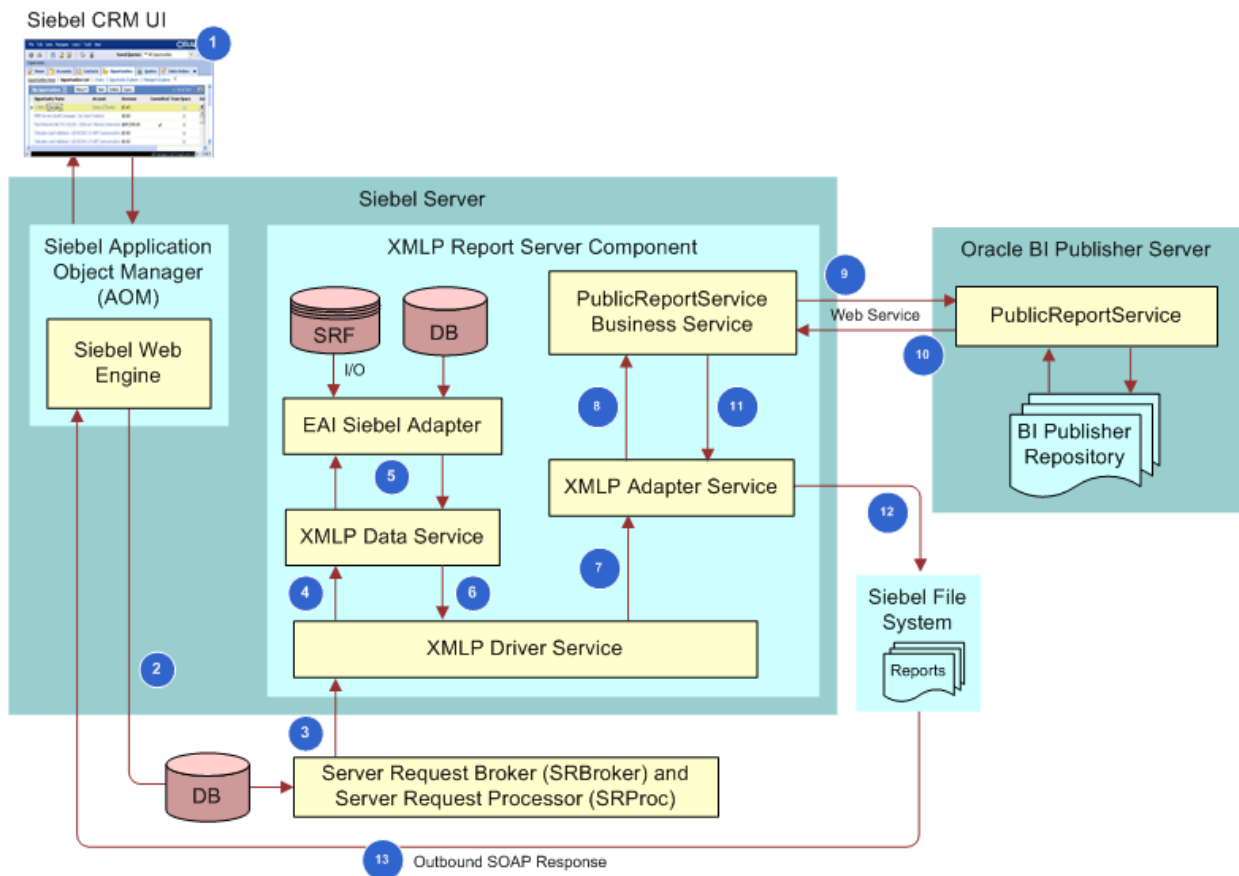


Figure 6. How Siebel CRM Runs Reports in Connected Clients

Explanation of Callouts

Siebel CRM does the following work when it runs a report in connected clients:

- 1 A user runs a report from the Run Report pane.

- 2 The Application Object Manager (AOM) sends an event to the Siebel Database.
- 3 The Server Request Broker (SRBroker) and the Server Request Processor (SRProc) monitor the request, and then sends it to the XMLP Report server component through the XMLP Driver Service.
- 4 The XMLP Driver Service calls the XMLP Data Service.
- 5 The XMLP Data Service gets data from the Siebel Database through the EAI Siebel Adapter, and then returns control back to the XMLP Data Service.
- 6 The XMLP Data Service sends control to the XMLP Driver Service.
- 7 The XMLP Driver Service calls the XMLP Adapter Service.
- 8 The XMLP Adapter Service calls the proxy `PublicReportServiceService` business service.
The `PublicReportServiceService` Web service is the interface between the Siebel application and the Oracle BI Publisher Server.
- 9 The `PublicReportService` business service makes a Web service call to the Oracle BI Publisher Server. For more information, see [“How Siebel CRM Uses the PublicReportServiceService Web Service” on page 80](#).
- 10 The Oracle BI Publisher Server runs the report, and then makes a Web service call to the `PublicReportServiceService` business service to return the report binary data.
The Oracle BI Publisher Server uses XSLT and XPath files to get XML data from the Siebel Database, and then merges this data with the report template that it gets from the Oracle BI Publisher repository. It then sends the report to the Siebel application. For more information, see [“How Siebel CRM Uses the Oracle BI Publisher Repository” on page 80](#).
- 11 The `PublicReportServiceService` business service returns control to the XMLP Adapter Service.
- 12 The XMLP Adapter Service downloads the report, and then creates the file in the Siebel File System. Siebel CRM displays the file in the Siebel application.

How Siebel CRM Uses the PublicReportServiceService Web Service

The `PublicReportServiceService` Web service is a Web service that Oracle BI Publisher uses to upload and send a report to the Siebel application. It does the following work:

- Validates privileges
- Gets information about reports, the Oracle BI Publisher repository, and the Oracle BI Publisher `ServerSharedFolder/SI EBELCRMREPORTS`
- Runs reports
- Creates and manages reports

How Siebel CRM Uses the Oracle BI Publisher Repository

The *Oracle BI Publisher repository* is a repository that contains predefined reports and new reports that you create. It resides on the Oracle BI Publisher Server. You install it when you install the Oracle BI Publisher Server. The following folder in the Oracle BI Publisher repository stores all reports:

SharedFolder/SIEBELCRMREPORTS

Siebel CRM stores the report template files that it requires to run a report in a separate folder in the SIEBELCRMREPORTS folder. For more information, see [“Directory Structure That Siebel Reports Uses” on page 86](#).

How Siebel CRM Runs Reports in Disconnected Clients

Figure 7 illustrates the architecture that Siebel CRM uses to run a report if the user is using a disconnected client. The Siebel Mobile Web Client and Siebel Developer Web Client allow you to run reports if disconnected from the Siebel Server.

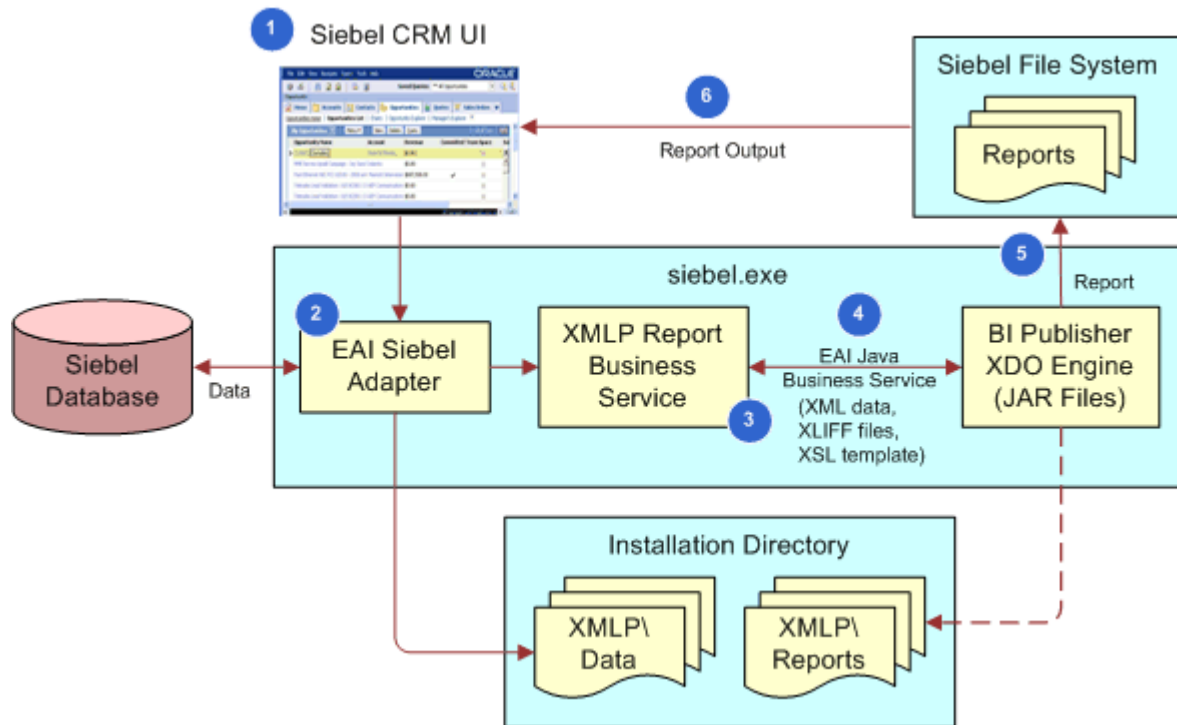


Figure 7. How Siebel CRM Runs Reports in Disconnected Clients

Explanation of Callouts

Siebel CRM does the following work when it runs a report in a disconnected client:

- 1 A user runs a report from the Run Report pane.
- 2 Siebel CRM uses the EAI Siebel Adapter to get data from the Siebel Database, and then stores this data as XML code in the following folder:

`SIEBEL_CLIENT_ROOT\CLASSES\XMLP\DATA`

- 3 Siebel CRM calls the XMLP Report Java Business Service.

- 4 The XMLP Report Business Service calls the EAI Java Business Service, and then loads the JAR files in the Java Virtual Machine.

The XMLP Report Business Service provides XML files to the Oracle BI Publisher XDO Engine to merge the report templates with XML data. One of the required inputs for the XDO Engine is an XML file. Siebel CRM uses the EAI Java Business Service to connect a Siebel application to the XDO Engine, and then uses the EAI Siebel Adapter to create the XML files. Oracle BI Publisher uses the EAI Java Business Service to send the XML file to the Oracle BI Publisher XDO Engine. This XML file contains the data that Siebel CRM uses to populate the report. Siebel CRM gets this data when it queries various data sources.

Siebel CRM merges the XML data file with the report template, and then the Oracle BI Publisher displays the report in the client after storing the file in the Siebel File System.

The XMLP Report Business Service and the Oracle BI Publisher libraries (XDO JAR files) are available as part of a Siebel mobile client installation. These libraries reside in the *SIEBEL_CLIENT_ROOT\CLASSES* folder. For information about installing a client, see *Siebel Installation Guide* for the operating system you are using.

The *EAI Java Business Service* is the JVM (Java Virtual Machine) interface that resides between the XDO Engine and the Siebel application. For more information about the EAI Java Business Service and Java Virtual Machine, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

- 5 The Oracle BI Publisher XDO Engine loads the XDO classes from the JAR files. Siebel CRM then sends the XML, XLIFF, and XSL templates to the XDO classes to run the report. Siebel CRM temporarily stores the report in the following folder, and then sends it to the Siebel File System:

SIEBEL_CLIENT_ROOT\CLASSES\XMLP\REPORTS

Oracle BI Publisher uses the XSLT and XPath files to get data from the XML data, and then incorporates this data into a report template file. The XSL data includes the XSLT, XPath, and XSL-FO standards that Siebel CRM uses to manipulate XML data. The XMLP Report server component uses the XDO Engine to convert the report template files into the XSL format. It then uses the BI Publisher-FO Processor to convert these files to the report that it publishes.

- 6 Siebel CRM sends control to the Siebel client, and then this client displays the report.

How Siebel Reports Uses Integration Objects

An *integration object* is a type of object that stores metadata that allows Siebel Reports to get Siebel data in XML format. It identifies the data that Siebel CRM must integrate. For example, a business object might contain multiple business components and fields, but most integrations only require some of these business components and fields. An integration object can specify only the business components and fields that Siebel CRM requires to run a report.

A typical Siebel integration uses Siebel EAI and integration objects to get data from an external application and then distribute this data to a Siebel deployment. Siebel EAI distributes this data in an integration message that includes header data. This header data identifies the message type, message structure, and a body that contains one or more instances of data. Example data includes orders, accounts, or employee records.

Siebel CRM uses this XML data to create a report template in RTF format and to run the report. Siebel CRM creates an XML data file that the following items use to merge the XML data with the report template, and then creates the report:

- **Connected client.** It uses the Siebel XMLP Report server component.
- **Disconnected client.** It uses the Oracle BI Publisher XDO Engine.

It is recommended that you modify an existing integration object before you create a new integration object. If no existing integration object meets your requirements, then you can create one. For more information, see [“Modifying Predefined Integration Objects to Add Fields to Reports” on page 101](#) and [“Creating New Integration Objects to Add Fields to Reports” on page 102](#).

For more information about integration objects, see *Integration Platform Technologies: Siebel Enterprise Application Integration*.

Computer Languages That Siebel Reports Uses

Siebel Reports uses the following file types:

- **XML (Extensible Markup Language).** A metalanguage that describes a markup language. It is a simplified version of SGML (Standard Generalized Markup Language). A markup language allows Siebel CRM to identify structures in a document. XML allows you to add markup to a document. It allows you to define tags and structural relationships between tags.
- **XSL (Extensible Stylesheet Language).** A language that defines style sheets. An XSL style sheet is a file that describes how to display an XML document of a given type. Siebel CRM uses XSL files only in a disconnected client. For more information, see [“How Siebel CRM Runs Reports in Disconnected Clients” on page 81](#).
- **XLIFF (XML Localization Interchange File Format).** A format that stores text and carries data from one step to another step when Siebel CRM must support multiple languages.
- **XSL-FO (Extensible Stylesheet Language-Formatting Objects).** An XML vocabulary that Siebel CRM uses to specify format semantics.
- **XSLT (Extensible Stylesheet Language Transformation).** A language that Siebel CRM uses to transform XML documents.
- **XPath (XML Path Language).** An expression language that XSLT uses to access or to refer to parts of an XML document.

For more information about the output types that a user can choose, see [“Output File Types That Siebel Reports Supports” on page 14](#).

About the Report Business Service

You can use the Report Business Service to run, share, or print a report automatically without requiring a user interaction. It allows Siebel CRM to run a report from a workflow process or in reply to a script that you write. You can create a workflow process that includes a business service step, and this step can reference a method of the Report Business Service. For example, you can create a workflow process that automatically does the following:

- Runs a report according to a specific query
- Saves a report in a specific format
- Sends a report to a customer in an email message

For more information about creating a workflow process, see *Siebel Business Process Framework: Workflow Guide*.

For more information about the Report Business Service and using business services with Siebel Reports, see the following documents on My Oracle Support:

- 1425724.1 (Article ID), Siebel BI Publisher Reports Business Service Methods
- 823360.1 (Article ID), Siebel BI Publisher Reports Business Service Methods
- 1362460.1 (Article ID), Using Oracle BI Publisher for Siebel Proposals

Guidelines for Modifying the Report Business Service

If you must modify the Report Business Service, then it is recommended that you use the following guidelines:

- Make sure you possess detailed knowledge and skills for the following items:
 - Using Siebel Tools
 - Using the Siebel Business Process Designer
 - Using the Oracle BI Publisher Server
 - Modifying a business service
 - Scripting
- Make sure Siebel BI Publisher is integrated. This integration allows Siebel CRM to use the business service methods of the Report Business Service.
- If you use the Report Business Service to upgrade from a prior release, then make sure you do the following work:
 - Make sure the following workflow processes are active:
 - BIP Report Generation
 - BIP Create Report Output
 - You can use the language code as an optional input argument in the XMLP Driver Service business service.

Report Business Service Methods

Table 6 describes the methods that you can use with the XMLP Driver Service of the Report Business Service.

Table 6. Methods That You Can Use with the Report Business Service

Method	Description
GenerateBIPReport	Runs a report in a connected client. This is the final method that you must use to run a report. It creates Siebel binary data, and then sends this data to BI Publisher server. It calls the RunBIPReport method to run the report.
GenerateReport	Runs a report in a disconnected client. This is the final method that you must use to run a report. It creates Siebel binary data, and then sends this data to BI Publisher server. It calls the RunBIPReport method to run the report.
EMailReport	Submits a scheduled report, and then emails this report when the report finishes. You must make sure you configure the email server that your deployment uses so that it uses the BI Publisher server to deliver the email message.
FTPReport	Submits a scheduled report, and then uses FTP to send this report when the report finishes. You must make sure you configure the FTP server that your deployment uses so that it uses the BI Publisher server to send the FTP.
FaxReport	Submits a scheduled report, and then faxes this report when the report finishes. You must configure the BI Publisher server so that it sends this fax to the fax computer.
LocalDeliveryReport	Saves a report in the BI Publisher repository that resides on the BI Publisher file server.
PrintReport	Submits a scheduled report, and then prints this report when the report finishes. You must make sure you configure the BI Publisher server to use this printer.
ScheduleReport	Schedules a report.
WEBDAVReport	Submits a scheduled report, and then sends this report to the domain according to the WEBDav that you configure on the BI Publisher server.
AttachEntity	<p>Sends input arguments to the Reports Business Service that determines how to do one of the following:</p> <ul style="list-style-type: none"> ■ Store the report in the Report Output business component ■ Attach the report to an Attachment business component. <p>The AttachEntity method is part of the GenerateBIPReport method.</p>

Directory Structure That Siebel Reports Uses

Table 7 describes the folders that Siebel Reports uses. These folders reside in the following folder on the Siebel Server:

SIEBEL_SERVER_ROOT\XMLP

Table 7. Siebel Reports Directory Structure

Directory	Description
<i>SIEBEL_SERVER_ROOT\XMLP\DATA</i>	Contains the XML files that Siebel CRM creates from integration objects. For more information, see “How Siebel Reports Uses Integration Objects” on page 82 .
<i>SIEBEL_SERVER_ROOT\XMLP\FONTS</i>	Contains the predefined .ttf (TrueType format) font files. For more information, see “Making Sure Fonts Are Available for Reports” on page 62 .
<i>SIEBEL_SERVER_ROOT\XMLP\REPORTS</i>	Contains the temporary files that Siebel CRM uses to run a report.
<i>SIEBEL_SERVER_ROOT\XMLP\TEMPLATES</i>	Contains the report template files. The user can download these report files to a local hard drive. This folder also contains the XSL files that a disconnected client requires. For information about how Siebel CRM uses these files to run a report, see “About the Siebel Reports Architecture” on page 77 .
<i>SIEBEL_SERVER_ROOT\XMLP\xliff\language_code</i>	<p>Contains the XLIFF files that Siebel CRM uses for multiple languages. Siebel CRM uses XLIFF files as the XML format that it uses to exchange data across multiple language. If Siebel CRM supports multiple languages, then it uses a separate language subfolder that resides in the xliiff folder for each language. It uses a language code to identify the language that this language subfolder contains. For example:</p> <ul style="list-style-type: none"> ■ XLIFF files for German reside in the <i>xliff\deu</i> folder. ■ XLIFF files for French reside in <i>xliff\fra</i> folder. <p>Siebel CRM registers a report template, and then stores the XLIFF file in an <i>SIEBEL_SERVER_ROOT\XMLP\language_code</i> folder on the Siebel Server. Siebel CRM places the XLIFF files for each language folder and the RTF file in a corresponding folder on the Oracle BI Publisher Server when you upload the report template.</p>

Directory Structure That the Oracle BI Publisher Repository Uses

Table 8 describes example folders that the Oracle BI Publisher Repository uses with a connected client.

Table 8. Siebel Reports Directory Structure

Example Directory	Description
ORACLE_HOME/xmlp/XMLP/Reports/SiebelCRMReports/Account List/	This directory contains the following files: <ul style="list-style-type: none"> ■ aclist.xdo. Contains the report definitions. ■ aclist.rtf. Contains the report template. ■ aclist_ja_JP.xlf. Example file of a language translation. ■ aclist_de_DE.xlf. Example file of a language translation.
ORACLE_HOME/xmlp/XMLP/Reports/SiebelCRMReports/Contact List/	This directory contains the following files: <ul style="list-style-type: none"> ■ cnlist.xdo. Contains the report definitions. ■ cnlist.rtf. Contains the report template. ■ cnlist_ja_JP.xlf. Example file of a language translation. ■ cnlist_de_DE.xlf. Example file of a language translation.

How Siebel CRM Controls Access to Reports

Siebel Reports use the same positions, responsibilities, and organizations to identify the reports a user can access that Siebel CRM uses to determine the views this user can access:

- The relationship that exists between the view and the report determines if a user can view a predefined report template.
- The position determines if the user can view a custom report template.
- The position determines if the user can share reports that finished running.

For more information about:

- Setting up responsibilities for reports, see the topic about setting up and enabling the Siebel Security Model for Siebel Reports in 1501378.1 (Article ID) on My Oracle Support.
- User access in relation to Siebel Reports, see the topic about security and authentication in 1501378.1 (Article ID), Integrating BI Publisher with Siebel Business Applications Version 8.1.1.x and Version 8.2.2.x, on My Oracle Support.
- Granting access to other users, see [“Sharing Report Templates” on page 107](#).
- General information about controlling access, see *Siebel Security Guide*.

How Siebel CRM Uses Report Templates With Organizations

The user can view reports that are specific to an organization, where the Active Position Organization equals the organization or organizations that you define for a custom report template. A user who can view more than one organization can change the Primary Active Position Organization field during a session.

For example, assume a sales representative and this representative's manager can view the same view, but the reports that these users can run vary depending on how Siebel CRM controls access to the reports. Assume the representative resides in the Holland organization, the manager resides in the Benelux organization, and that Benelux is the parent organization of the Holland and Belgium organizations. When accessing the Run Report pane:

- The sales representative and the manager can view any report that is defined in the report template views and that is associated with the view.
- The sales representative can view custom templates that:
 - The representative created in the My Templates view. These templates include only a default organization.
 - Are explicitly associated with the Holland organization, assuming the position is associated with the Holland organization.
- The manager can view custom templates that:
 - The manager created in the My Template View or templates his subordinates created that appear in the My Teams view. These templates include only a default organization.
 - Are explicitly associated to the Benelux organization, given his current active position organization is Benelux.
 - A user created, where this user is associated with the Holland or Belgium organizations. The manager can view any report template that a subordinate creates when associated with any organization.

7

Customizing Siebel CRM Reports

This chapter describes how to customize reports. It includes the following topics:

- [Process of Creating Custom Reports on page 90](#)
- [Specifying the Siebel CRM Data That Report Templates Use on page 97](#)
- [Defining Optional Fields When Registering Report Templates on page 105](#)
- [Customizing Reports That Use Parameters on page 107](#)
- [Customizing Other Options for Siebel Reports on page 114](#)

Process of Creating Custom Reports

To create a custom report, do the following tasks:

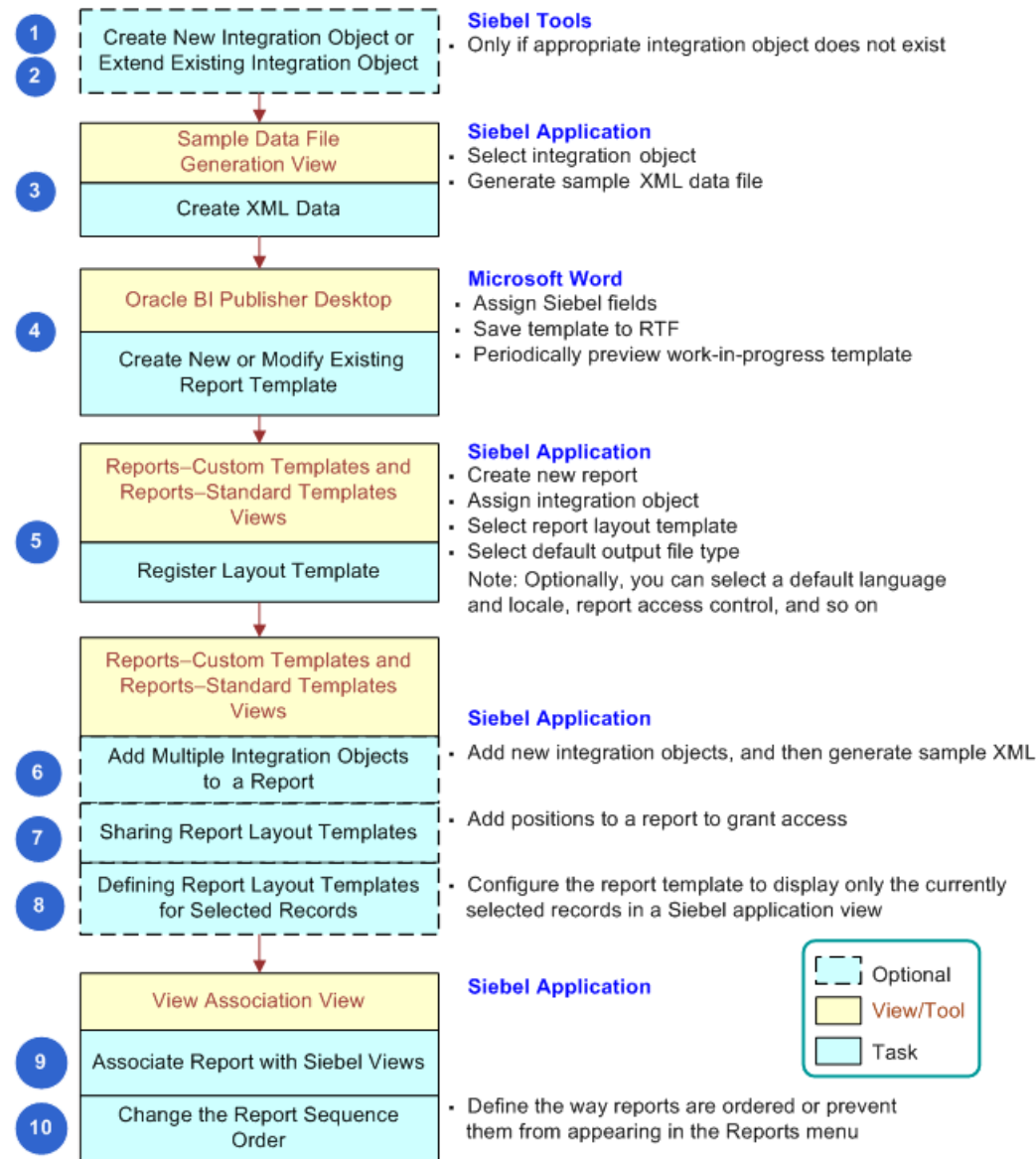


Figure 8. Process of Creating Custom Reports

- 1 "Creating Report Templates" on page 91
- 2 "Registering Report Templates" on page 93
- 3 "Associating Report Templates with Views" on page 97

Creating Report Templates

This topic describes how to use Oracle BI Publisher Desktop to create a report template. For more information, see [“About Oracle BI Publisher Desktop” on page 15](#).

This task is a step in [“Process of Creating Custom Reports” on page 90](#).

To create report templates

- 1 Determine if you can use a predefined report template.

It is recommended that you create a new report template only if no predefined report template meets your report layout requirements. The following are some examples of when you can use a predefined template:

- You can add new fields to an existing report template.
- Differences between your new report and an existing report are minor.
- You require multiple versions of the same report where each version displays slightly different data to different users.

The following are some situations where you must create a new report template:

- The report requires a new integration object.
- The report requires a new integration component. For example, you create a new report for a view that references the same business object.

For more information, see [“How Siebel Reports Uses Integration Objects” on page 82](#).

- 2 Install Oracle BI Publisher Desktop.

NOTE: It is recommended that you use the same version of Oracle BI Publisher Desktop as the Oracle BI Server you are running. For example, if you are using Oracle BI Publisher Desktop 11, then it is recommended that you run it with Oracle BI Publisher Server 11g.

For detailed information about installing Oracle BI Publisher Desktop and creating a report template, see *Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher*. In the Oracle Business Intelligence Publisher Documentation Library available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>). Siebel CRM does not support all features that the Oracle Business Intelligence Publisher documentation describes. It only supports features that this guide describes.

- 3 Start Microsoft Word.
- 4 Choose the Oracle BI Publisher menu, Data, and then the Load Sample XML Data menu item.
- 5 In the Select XML Data window, choose an XML data file.

XML data files reside in the following folder on the Siebel Server:

```
SIEBEL_SERVER_ROOT\XMLP\DATA
```

For more information, see [“Directory Structure That Siebel Reports Uses” on page 86](#).

- 6 Use the Template Wizard to define the report template:

- a In the Microsoft Word toolbar, choose Insert, Table/Form, and then Wizard.
- b In the Template Wizard, choose one of the following report formats, and then click Next:
 - Table
 - Form
 - Free Form
- c (Optional) Change the XML data group, and then click Next.
- d Choose the Siebel fields that the report must display, and then click Next.
- e If the report must display data in Table format, then specify the group and sort requirements, and then click Next.
- f Add labels for the fields that the report displays, and then click Finish.
- g Save the report template in RTF format.

Siebel CRM creates the following files in the Siebel File System:

- The RTF file for the report template and the XLIFF files for a connected client.
 - The XSL and XLIFF files for a disconnected client.
- h Open the RTF report template in Microsoft Word using the BI Publisher Add-on.
 - i In Microsoft Word, click Translation, and select Extract Text.

The report template is saved as an XLIFF file.

7 Preview the report template:

- a In Microsoft Word, open the template file.

Siebel CRM stores the template files in the following folder:

SI EBSRVR_ROOT\XMLP\TEMPLATES

- b Choose the Oracle BI Publisher menu, Preview Template, and then choose the output file format. Siebel CRM displays the report in the format that you choose.
- c Preview different output file formats.

8 Copy the report template files to the Siebel Server:

- If you are using a disconnected client, then copy the following files from this client to the Siebel Server:
 - Copy Report template files that are in RTF format (.rtf) to the *SI EBSRVR_ROOT\XMLP\Templates* folder.
 - Copy XLIFF (.xlf) files to the *SI EBSRVR_ROOT\XMLP\xliff\language_code* folder.
- If you are using a connected client, then save the template file in RTF format, and then copy this RTF file to the Siebel Server in the following folder:

`SI EBSRVR_ROOT\XMLP\TEMPLATES`

Make sure you do not include any special characters in the report name. For more information about these folders, see [“Directory Structure That Siebel Reports Uses” on page 86](#).

Registering Report Templates

You must register a report template so that the user can choose the report in the Run Report pane.

You can associate each report with only one report template, but you can register the same report template with many reports.

This task is a step in [“Process of Creating Custom Reports” on page 90](#).

To register a report template

- 1 Make sure the Oracle BI Publisher Server is running:
 - a In the computer where the Oracle BI Publisher Server is installed, click Start, Programs, and then the Oracle - BIPHomeX menu item.
 - b Choose Start BI Publisher.

A command window displays. You must leave this command window open while the Oracle BI Publisher Server is running.
- 2 Log in to the Siebel client with administrator privileges, and then navigate to the Administration - BIP Publisher Reports screen.
- 3 In the template list, click New.

For more information, see [“Views You Use to Register Report Templates” on page 94](#).
- 4 Enter a name in the Report Name field and an optional description in the Description field.

If you enter the name of a report that already exists, then Siebel CRM displays an error message. You must enter a unique name.
- 5 In the Integration Objects list, specify the Siebel CRM data that the report template uses.

Ignore the Primary Integration Object Name field. Siebel CRM automatically enters a value in this field when you choose an integration object in the child Integration Objects list. For more information, see [“Specifying the Siebel CRM Data That Report Templates Use” on page 97](#).
- 6 Choose the report template file that Siebel CRM uses to run the report:
 - a In the Template field, click the Magnifying Glass.
 - b In the Add Attachment dialog box, click Choose File and then navigate to the following folder:

`SI EBSRVR_ROOT\XMLP\TEMPLATES`

Siebel CRM stores report template files in this folder. For more information, see [“Directory Structure That Siebel Reports Uses” on page 86](#).

- c Choose the report template file that this report must use, and then click Add.

For example, choose the file you saved in [Step 8 on page 92](#).

- 7 Define the optional fields, as necessary.

For example, you can choose a start date in the Start Date field, and an end date in the End Date field to specify the time period when the user can run the report. You can also specify the default output type, how to do multiple language conversion, and so forth. For more information, see [“Defining Optional Fields When Registering Report Templates” on page 105](#).

- 8 Upload the files:

- If you use a connected client, then click Upload Files.
- If you use a disconnected client, then locate and upload the RTF and XLIFF files.

Siebel automatically generates the XSL file and uploads these three files to the Siebel Server and BI Publisher XDO Engine.

For more information about how to upload files, see [“How Siebel CRM Uploads Report Templates” on page 95](#).

- 9 Verify that Siebel CRM uploaded the report:

- a Log in to the Oracle BI Publisher Server with administrator privilege.
- b Click the Reports tab, expand the Shared Folders tree, and then click SiebelCRMReports.
- c Verify that the new report template and associated files were uploaded.

Views You Use to Register Report Templates

The Administration - BI Publisher Reports screen includes the following views that you use to modify and register a report template:

- **Reports - Custom Templates.** Lists custom report templates that Siebel CRM filters according to Position. If you can view the My Templates list in the Reports - Custom Templates view, then you can create your own custom report template. Only you can view or run a custom report template that you create. You can share a report template with another user if this user can access the view that you associate with this report in the Siebel client. The All Templates Across Organizations list allows a manager or administrator to view the report templates that a subordinate creates.
- **Reports - Standard Templates.** Lists predefined and custom report templates that are available across all organizations, but not according to Position. To prevent a user from creating a report template and then running it, you can disallow access to the views that the template references. For example, you can limit access to the Reports - Standard Templates view to only administrators. For more information, see [“How Siebel CRM Controls Access to Reports” on page 87](#).

CAUTION: It is strongly recommended that you limit access to standard templates only to administrators and developers. If you do not do this, then a user might mistakenly modify a report template that all users can access. You can use Siebel responsibilities to limit access to this view, and you can assign the XMLP_DEVELOPER responsibility only to users who must access custom templates.

How Siebel CRM Uploads Report Templates

Figure 9 illustrates how Siebel CRM uploads the files associated with a report template when you register this template. For more information, see “About Oracle BI Publisher Desktop” on page 15.

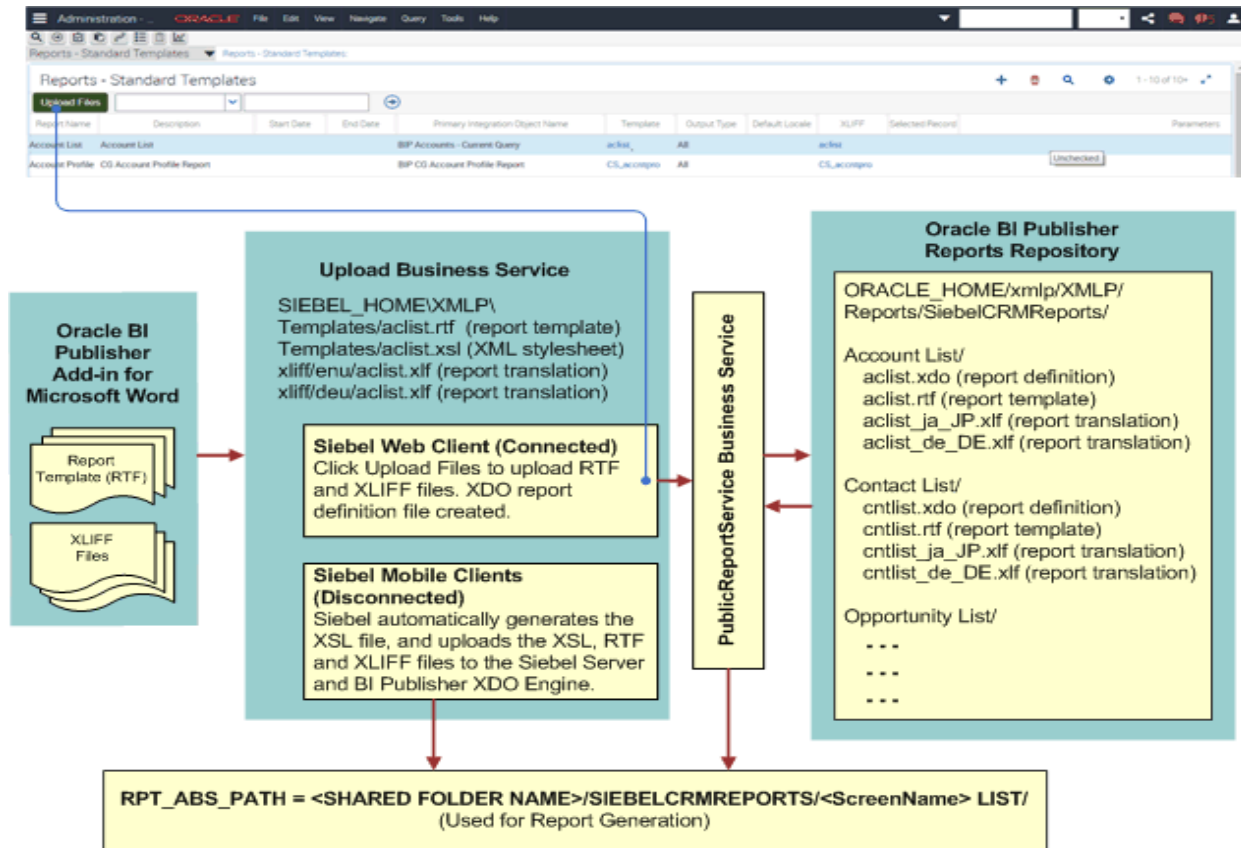


Figure 9. How the Report Template and Translation Files Are Uploaded from Oracle BI Publisher Desktop to Run Reports

Figure 9 illustrates the following:

- 1 You use Oracle BI Publisher Desktop to create a report template, and then do one of the following:
 - a In a connected client, you click the Upload Files button in the templates list of the Administration - BIP Publisher Reports screen, and then Siebel CRM does the following work:
 - Uses the PublicReportServiceService Web service to do the upload.
 - Creates a new folder structure in the Oracle BI Publisher repository.

- Uploads the RTF and XLIFF files to this new folder structure. This folder uses the same name that the uploaded report uses, and it contains the XLIFF and .xdo report template files. If the RTF and XLIFF files do not exist in this folder structure, then Siebel CRM cannot create the report.

The Oracle BI Publisher Server creates the XSL file and the XDO report definition file when the user runs the report from the Run Report pane.

Siebel CRM disables the Generate XLIFF button in a connected client.

For more information, see [“How Siebel CRM Runs Reports in Connected Clients” on page 79](#).

- b In a disconnected client, you must locate and upload the RTF and XLIFF files because Siebel CRM disables the Upload Files button.

Siebel automatically generates the XSL file and uploads the XSL, RTF, and XLIFF files to the Siebel Server and BI Publisher XDO Engine.

Siebel CRM stores report templates and XLIFF files in the Siebel File System and synchronizes them to the Siebel Server when a Mobile Client does a synchronization.

All RTF, XLIFF, and XSL files are local to the Oracle BI Publisher XDO Engine when Siebel CRM runs the report. For more information, see [“How Siebel CRM Runs Reports in Disconnected Clients” on page 81](#).

- 2 In a connected client, Siebel CRM uses the PublicReportServiceService Web service to store the RTF and XLIFF files in the Oracle BI Publisher repository.
- 3 In a disconnected client, Siebel CRM stores the report templates and XLIFF files in the following folders:

```
SIEBEL_ROOT_CLIENT\xml p\templates  
SIEBEL_ROOT_CLIENT\xml p\XLIFF
```

Validations That Siebel CRM Does When You Upload a Report Template

Siebel CRM does the following work when you associate an RTF, XLIFF, or ZIP file with a report template during the upload:

- Makes sure the report template does not already use the RTF file name.
- Makes sure the Template field includes the name of an RFT file.
- Makes sure the XLIFF file name is the same name as the template file name.
- Makes sure the XLIFF field contains the name of an XLF or ZIP file.

Associating Report Templates with Views

This topic describes how to associate a report template with a view. Siebel CRM displays different reports in the Report Name list of the Run Report pane depending on the view that you associate with the report. For more information, see [“How Siebel CRM Displays Reports According to the View” on page 14](#).

NOTE: When you associate a report template to a view, the business object on which the view is based needs to be the primary business object for the integration object.

This task is a step in [“Process of Creating Custom Reports” on page 90](#).

To associate report templates with views

1 Navigate to the Administration - BIP Publisher Reports screen, and then the View Association view.

2 In the Views list, choose a view.

The Run Report pane displays the report in the Report Name dropdown list when the user navigates to this view. You add this report in [Step 4](#).

3 In the child Report List, click Add.

4 In the Reports dialog box, choose a report name, and then click OK.

5 (Optional) In the child Report List, enter a number in the Sequence field.

For more information, see [“Modifying the Sequence of Report Names in the Run Report Pane” on page 114](#).

6 Test the new report.

For more information, see [Chapter 3, “Running Reports.”](#)

Specifying the Siebel CRM Data That Report Templates Use

This topic describes how to add integration objects to report templates. It includes the following information:

- [“Modifying Integration Objects” on page 100](#)
- [“Creating XML Files from Integration Objects” on page 103](#)
- [“Reducing the Amount of Data That Integration Objects Transfer” on page 104](#)

You add an integration object to instruct Siebel CRM where to get the data that it displays in a report, such as data from a business component field. For more information, see [“How Siebel Reports Uses Integration Objects” on page 82](#).

To specify the Siebel CRM data that report templates use

- 1 Log in to the Siebel client with administrator privileges, and then navigate to the Administration - BIP Publisher Reports screen.
- 2 Choose a report in the template list.
For more information, see [“Views You Use to Register Report Templates” on page 94](#).
- 3 In the Integration Objects list, click New.
- 4 Specify the integration object, using values from the following table.

Field	Description
Primary	Indicates that this integration object is the primary integration object. Siebel CRM sets the first integration object that you add as the primary integration object.
Integration Object Name	Choose the integration object that you must associate with the report template. Siebel CRM displays the predefined and custom integration objects that currently exist in the Siebel runtime repository. If no integration object exists that meets your requirements, then see “Modifying Integration Objects” on page 100 . NOTE: When working with multiple repositories, select the Repository Name field in the Repository Integration Object business component and add it to the Integration Objects list. This facilitates integration of integration object association to a report template.
Search Specification	Enter a search specification that Siebel CRM uses to filter the data that this report displays. If you create a nonprimary integration object, then you must enter a value in the Search Specification field. For more information, see “Specifying Search Specifications with Multiple Integration Objects” on page 99 .

- 5 (Optional) Examine an example of the report output:
 - a Click Generate Sample XML.
 - b In the file download dialog box, choose Open to view the report.

Siebel CRM displays an example of the report output. For example if you specify the BIP Account List integration object, and then click Generate Sample XML, then Siebel CRM displays a file download dialog box that allows you to open or save the BIP Account List.xml file.

Siebel CRM creates one XML data file for multiple integration objects. To create XML for only one integration object, see [“Creating XML Files from Integration Objects” on page 103](#).

Specifying Search Specifications with Multiple Integration Objects

You can customize a report so it that it runs across multiple business objects. To do this, you use multiple integration objects, where one integration object is the primary and all others are nonprimary integration objects.

If you specify an integration object in the report template in the Integration Objects list, then the search specification property of every nonprimary integration object must include a valid search specification. For example, you can use the following format:

```
'Integration_Component_Name'. Search = "[Field_Name] = ""value_to_filter""
```

For example:

```
'Contact'. Search = "[Last Name] = ""Sh""
```

Siebel CRM sends the query that a user runs in a view to the primary integration object. It does not send this query to a nonprimary integration object. It requires a valid search specification so that it can filter the data for nonprimary integration objects. If a nonprimary integration object does not include a valid search specification, then this integration object returns all data from the business components, including unwanted data, and it might degrade performance.

It is recommended that you test these search specifications to make sure they are valid. For more information about search specifications, see *Configuring Siebel Business Applications*.

How Siebel CRM Applies Search Specifications to Filter Reports

The following items can filter the records that Siebel CRM displays in a report:

- Client query (the user enters a query in a field)
- Predefined query
- Search specification on the business component
- Search specification on the applet
- Search specification on the integration object

Table 9 describes different examples Siebel CRM applies search specifications to filter reports. In all rows, assume that the applet and the business component each include a search specification. For example, the first row describes how Siebel CRM filters records if the user enters a query in the client, does not use a predefined query, there is no search specification for the integration object, and the applet and the business component each include a search specification. For more information about creating search specifications, see *Configuring Siebel Business Applications*.

Table 9. Integration Object Search Specification Run-Time Behavior for Siebel CRM

Client Query	Predefined Query	Integration Object	Filters That Siebel CRM Applies
Y	N	N	Applies the UI query, applet search specification, and business component search specification.
N	Y	N	Applies the predefined query, applet search specification, and business component search specification.

Table 9. Integration Object Search Specification Run-Time Behavior for Siebel CRM

Client Query	Predefined Query	Integration Object	Filters That Siebel CRM Applies
Y	N	Y	Applies the applet search specification and business component search specification. Appends the integration object search expression and overrides the UI query. For more information about search expressions and sort specifications, see <i>Configuring Siebel Business Applications</i> .
N	Y	Y	Applies the applet search specification and business component search specification. Appends the integration object search expression and overrides the predefined query.

The following items apply for each row in [Table 9](#):

- The Position automatically preserves visibility, organization visibility, and view mode. For information, see [“How Siebel CRM Controls Access to Reports” on page 87](#).
- Using the CTRL key to choose records does not affect query behavior. For more information, see [“Using the CTRL Key to Choose Multiple Records for a Report” on page 22](#).
- Mobile Web Client functionality is not adversely affected.
- Multilingual behavior does not affect the functionality.

Modifying Integration Objects

This topic describes how to modify a predefined integration object or create a custom integration object so that you can add fields to a report. You can do the following after you modify or create a new integration object:

- Use the integration object when you register a report template. For more information, see [“Registering Report Templates” on page 93](#).
- Use the integration object in the Sample Data File Generation view to create a custom report. For information, see [“Creating XML Files from Integration Objects” on page 103](#).

CAUTION: If you modify a predefined integration object or create a custom integration object, then it is strongly recommended that you get help from someone who is familiar with using Siebel Tools and configuring Siebel Business Applications. For more information, see *Integration Platform Technologies: Siebel Enterprise Application Integration*. For more information, see [“How Siebel Reports Uses Integration Objects” on page 82](#).

Modifying Predefined Integration Objects to Add Fields to Reports

Siebel CRM comes with predefined reports and each of these reports references an integration object that provides the report schema. This topic describes how to modify an existing integration object so that Siebel CRM can use it to add fields to a report. If necessary, you can reduce the size of the integration object that you use to improve performance. For more information, see [“Reducing the Amount of Data That Integration Objects Transfer” on page 104](#).

To modify predefined integration objects to add fields to reports

- 1 Log in to the Siebel client, and then identify the integration object that you must modify:
 - a Navigate to the Administration - BIP Publisher Reports screen.
 - b In the template list, choose the report where you must add a new field.
For more information, see [“Views You Use to Register Report Templates” on page 94](#).
 - c Note the value that Siebel CRM displays in the Primary Integration Object Name field.
Siebel CRM prefixes the integration objects that it uses with BIP.
- 2 Add a new field to the integration object:
 - a Log in to Siebel Tools.
 - b In the Object Explorer, click Integration Object.
If the Object Explorer does not display the Integration Object type, then do the following:
 - Click the View menu, and then the Options menu item.
 - In the Development Tools Options dialog box, click the Object Explorer tab.
 - Make sure the integration object and all child object types of the integration object type contains a check mark, and then click OK.
 - c In the Integration Objects list, locate the integration object that you identified in [Step 1](#).
 - d In the Object Explorer, expand the Integration Object tree, expand the Integration Component tree, and then click Integration Component Field.
 - e Scroll through the Integration Component Field list until you locate the field that the report must display.
 - f Note the values in the following properties:
 - External Name
 - External Data Type
 - External LengthEach of these properties reference a property of the business component field that contains the data that the report must display. For example, the External Name of the integration object field references the Name property of the business component field.
 - g In the Integration Component Field list, create a new integration component field:
 - Enter the values you noted in [Step f on page 101](#) into the External Name, External Data Type, External Length properties.

- ❑ Set the XML Sequence property.
- ❑ Set the XML Tag property. You typically set the XML Tag property to the same value that the Name property contains.

Siebel CRM uses these XML properties when it uses the XML file to get data from the Siebel Database. You must use the syntax that an XML tag requires. You must not include any spaces or special characters. The following prefix is not required:

SS_

- 3 Reduce the Size of the integration object that you modified in [Step 2](#).

For more information, see [“Reducing the Amount of Data That Integration Objects Transfer” on page 104](#).

- 4 Deliver your changes.
- 5 Migrate the Integration Object changes through the Siebel Migration Application using the Application Incremental Runtime Repository Data Service.

Creating New Integration Objects to Add Fields to Reports

Siebel CRM populates a QueryString field in the sample XML and parses it as a field value in the integration object while the report runs, by default. This field contains the user-defined query that the user entered to run the report. You can display this field value on the report template. For example, the format of the field might be [Account Status] = “Active”].

To create a new integration object to add fields to reports

- 1 Make sure no predefined integration objects exist that meet your report layout requirements.
Siebel CRM comes with predefined integration objects. It is recommended that you create a new integration object only if the predefined integration objects do not meet your requirements. For more information, see [“Modifying Predefined Integration Objects to Add Fields to Reports” on page 101](#).
- 2 Log in to Siebel Tools.
- 3 In the Object Explorer, click Integration Object.
- 4 Right-click in the Integration Objects list, click New Objects Wizard, and then click Integration Object.
- 5 Use the Integration Object Builder wizard to create the new integration object.

Note the following:

- Make sure you prefix the name of this new integration object with BIP. You must add the BIP prefix so that Siebel CRM displays this integration object in the Sample Data File Generation view in the client. For information about this view, see [“Creating XML Files from Integration Objects” on page 103](#).
- Make sure you include in the integration component any fields that the user can query in the applet that references the master business component. For more information, see [“About Master-Detail Reports” on page 115](#).

- 6 Reduce the Size of the integration object that you created in [Step 5](#).

For more information, see ["Reducing the Amount of Data That Integration Objects Transfer" on page 104](#).

- 7 Deliver your changes.

- 8 Use the Siebel Migration Application Incremental Runtime Repository Data Service to migrate integration object changes to your target environment.

For more information about Siebel Migration and the Migration Application Incremental Runtime Repository Data Service, see *Siebel Database Upgrade Guide*.

Creating XML Files from Integration Objects

This topic describes how to create the XML files that Siebel CRM uses to run a report. You use the Sample Data File Generation view to choose the integration object that provides the Siebel CRM data that Siebel CRM displays in a report.

To create XML files from integration objects

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - BIP Publisher Reports screen, and then the Sample Data File Generation view.
- 3 In the Sample Data File Generation list, choose an integration object.
- 4 Click Generate Sample XML.

Siebel CRM creates an XML file and names it using the following format:

integration_object_name.xml

For example:

BIPLiteratureFullfilment.xml

- 5 Save the file in the following folder:

SI EBSRVR_ROOT\XMLP\DATA

You can now create a report template that uses this XML data. For more information, see ["Creating Report Templates" on page 91](#).

Creating XML Files from Integration Objects for All Files

Siebel CRM saves only some of the input and output data when it creates a sample XML data file. Oracle BI Publisher Desktop might not display fields that do not contain data, such as a multivalue field. This topic describes how to create XML data for all files, including these files that Oracle BI Publisher Desktop might not display.

To create XML files from integration objects for all files

- 1 Open Siebel Tools.
- 2 In the Object Explorer, click Integration Object.
- 3 In the Integration Objects list, locate an integration object.
- 4 In the Integration Objects list, click Generate Schema.
- 5 In the Generate XML Schema dialog box, choose a business service, envelope type, and a file name that you can locate in Windows Explorer, and then click Finish.
- 6 Use Windows Explorer to locate the file you specified in [Step 5](#), and then rename the file with an .xsd extension.
- 7 Use the Oracle BI Publisher menu in Microsoft Word to load the file you renamed in [Step 6](#).

All fields are now available for use in a report template. For more information, see [“About Oracle BI Publisher Desktop” on page 15](#). For more information about creating an XML schema, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Reducing the Amount of Data That Integration Objects Transfer

The number of integration components and integration component fields that you add can degrade performance, particularly in a report that includes a large amount of data.

It is recommended that you review and configure all business components and integration components; this is to prevent unnecessary scripting execution during report generation. It is also recommended to include only the fields in an integration object that Siebel CRM requires to run the report. A large integration object increases the time that the XMLP Report server component requires to get data through the Siebel Enterprise Application Integration (EAI), and it increases the size of the XML that the Oracle BI Publisher Server uses to run the report. If a report template includes logic to get, aggregate, or reference elements in the XML, then a large integration object can also affect performance when Siebel CRM displays the report.

To reduce the amount of data that integration objects transfer

- Create a separate integration object for each report.
Do not create a single, large integration object that multiple reports use.
- Deactivate every integration component field that the report does not require.
If you use the EAI Siebel Wizard to create an integration object, then it adds all business component fields to the integration component. You must deactivate fields that the report does not require after the wizard finishes running.
- Do not flag all business component fields as Force Active. When a business field is flagged as Force Active, this loads the business component field during each query. If a minimal number of fields are flagged as Force Active, this improves performance.

Defining Optional Fields When Registering Report Templates

This topic describes how to define the optional fields when you register a report template. It includes the following information:

- [“Allowing Users to Choose The Records That a Report Contains” on page 106](#)
- [“Sharing Report Templates” on page 107](#)

To define optional fields when registering report templates

- 1 Specify the start date and end date:
 - a Choose a start date in the Start Date field, and then click Done.

The start and end dates specify the time period when the user can run the report. A predefined report does not include a value in the Start Date field. Siebel CRM automatically enters a value in the Start Date field for a custom report that you create. You can change this value at any time:

 - If you choose a start date, then you must also choose an end date.
 - If you choose an end date, then you must also choose a start date.
 - b Choose an end date in the End Date field, and then click Done.

Siebel CRM no longer displays the report in the Run Report pane after the end date occurs. If you leave the End Date field empty, then Siebel CRM will always display the report.
- 2 Configure the default Output Type. In the Output Type field, choose the output format that Siebel CRM automatically uses when it saves the report.

You typically set this value to All. If you choose only one output type, then Siebel CRM does not display the Report Output Type dialog box when the user chooses this report in the Run Report pane. Instead, it saves the report using the value that you choose in the Output Type field. For more information, see [“Output File Types That Siebel Reports Supports” on page 14](#).
- 3 Configure the default language for the report template:
 - Set the default language that Siebel CRM uses when it runs a report that uses this report template.

For more information, see [“Priority That Determines the Report Language” on page 72](#) and [“Creating Multilingual Reports” on page 73](#).
 - a Choose the XLIFF file that contains the strings that Siebel CRM uses for column names, field names, and captions in the report:
 - In the XLIFF field, click the Magnifying Glass.
 - In the Add Attachment dialog box, click Choose File and then navigate to the following folder:

`SI EBSRVR_ROOT\XMLP\TEMPLATES`

Siebel CRM stores XLIFF files in this folder. For more information, see [“Directory Structure That Siebel Reports Uses” on page 86](#).

You must specify an XLIFF file so that Siebel CRM can register a report template even if this report template is not multilingual.

- Choose the XLIFF file that this report must use, and then click Add.

4 Specify the following optional fields:

- a Add a check mark to the Selected Records field to allow users to choose records.

For more information, see [“Allowing Users to Choose The Records That a Report Contains” on page 106](#).

- b Specify how to share this report template:

- Use the Report Access field to specify the positions that can access this report template or share report output.
- Use the Organization field to specify the organizations that can access this report template.

These fields are available only in the Reports - Custom Templates view. For more information, see [“Sharing Report Templates” on page 107](#).

5 Add report parameters.

For more information, see [“Customizing Reports That Use Parameters” on page 107](#).

Allowing Users to Choose The Records That a Report Contains

You can configure a report template to include only the records that the user chooses in a view. If you encounter problems using this feature, then you might need to do some configuration in Siebel Tools. For more information, see 1180903.1 (Article ID) on My Oracle Support.

To allow users to choose the records that a report contains

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - BIP Publisher Reports screen.
- 3 Locate the report template you must modify.

For more information, see [“Views You Use to Register Report Templates” on page 94](#).

- 4 In the Selected Records field, do one of the following:

- **Add a check mark.** The report will include only the records that the user chooses in the list or detail applet in the client. The report will also include records that reside in child views and grandchild views of each record that the user chooses in the parent list. For information about how the user chooses multiple records, including important caution information, see [“Using the CTRL Key to Choose Multiple Records for a Report” on page 22.](#)
- **Remove the check mark.** The report will include all the records that Siebel CRM displays in the list or detail view.

Sharing Report Templates

This topic describes how to share a report template across positions and organizations. For more information about how Siebel CRM uses positions and organizations, see [“How Siebel CRM Controls Access to Reports” on page 87.](#)

To share report templates

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - BIP Publisher Reports screen, and then the Reports - Custom Templates view.
- 3 In the templates list, locate the report template you must modify.
- 4 (Optional) Specify the positions that can access the report:
 - a Click the Report Access field.
The Report Access field is available only in the Reports - Custom Templates view. For more information, see [“Views You Use to Register Report Templates” on page 94.](#)
 - b In the Position dialog box, choose the positions that must access the report, click Add, and then click OK.
The Report Access field determines the users who can access the report template according to position. If a user is associated with the position you add, then this user can access this report template or share report the output that Siebel creates when it uses this template. You can use the CTRL key to choose multiple positions.
- 5 (Optional) Specify the organizations that can access the report:
 - a Click the Organization field.
 - b In the Organizations dialog box, choose the organization that must access the report, click Add, and then click OK.

Customizing Reports That Use Parameters

This topic describes how to customize a report that uses parameters. It includes the following information:

- [“About Report Parameters” on page 108](#)
- [“Adding Report Parameters to Report Templates” on page 108](#)
- [“Report Parameter Types You Can Specify” on page 111](#)
- [“Adding Scrollbars to the Parameters Section of the Run Report Pane” on page 112](#)
- [“Guidelines for Defining Parameters in Report Templates” on page 113](#)

About Report Parameters

NOTE: A *report parameter* is a type of filter that refines the data Siebel CRM sends to a report template file that Oracle BI Publisher uses to run a report. It allows the user to refine the report output according to data that the user might not be able to query in a view. Report parameters allow you to do the following work:

- Narrow the query, sorting, or grouping when running a report.
- Produce many different reports from the same report template.

The user uses the Parameter section in the Run Reports pane to add report parameters. Siebel CRM stores the parameters that the user chooses and the parameter values that the user enters in temporary memory in the client. Siebel CRM removes these items from memory after the user clicks Submit.

Siebel Open UI supports report parameters starting with Siebel CRM versions 8.1.1.10 and 8.2.2.3.

For information about how to run a report that uses parameters, see [“Using Report Parameters to Filter Reports” on page 29](#).

If you must use report parameters in a scheduled report, then you must install an Oracle BI Publisher patch. For information about this patch, see 880452.1 (Article ID), *Siebel Maintenance Release Guide* on My Oracle Support. For more information about scheduling a report, see [“Scheduling Reports” on page 25](#).

Adding Report Parameters to Report Templates

This topic describes how to define report parameters so that Siebel CRM allows the user to choose these parameters in the Run Report pane.

To add report parameters to report templates

- 1 Add the report parameter in Oracle BI Publisher Desktop.
For more information, see [“Adding Report Parameters in Oracle BI Publisher Desktop” on page 110](#).
- 2 Log in to the Siebel client with administrator privileges.

- 3 Navigate to the Administration - BIP Publisher Reports screen, and then the Reports - Custom Templates view.

For more information, see [“Views You Use to Register Report Templates” on page 94](#).

- 4 In the Templates list, choose a template, and then add a check mark to the Parameters field. This field makes sure Siebel CRM displays the Parameters section in the Run Reports pane.

- 5 In the Parameters list, click New.

To avoid a problem, it is recommended that you use the New button instead of Copy Record.

- 6 Specify values for the report parameter, using information from the following table.

Field	Description
Label	Enter text that describes the report parameter. Siebel CRM displays the text you enter in the Parameter section of the Run Report pane.
Name	Enter a name for the report parameter. The value you enter field must reference a parameter definition that exists in the report template. For example, if you set the Name field to Param1, then the report template must include a definition for Param1.
Order	Enter a number that indicates the sequence that Siebel CRM uses to display report parameters in the Parameter section of the Run Report pane. Siebel CRM sequences these report parameters in ascending numeric order. For example, if the Parameters list includes Param1 and Param2, and if you set the Order for Param1 to 1 and Param2 to 2, then Siebel CRM displays Param1 immediately before Param2. If the Order field is empty, then Siebel CRM displays this report parameter after the report parameters that include a value in the Order field.
Type	Specify the type of parameter. For more information, see “Report Parameter Types You Can Specify” on page 111 .
Default	Enter the default value that Siebel CRM displays in the Run Report pane for this parameter.
Picklist Name	Enter the name of the picklist that allows the user to choose a value. Use this field only if you set the Type field to LOV.
Required	Add a check mark to the Required field to require the user to set this parameter.

- 7 (Optional) If Siebel CRM must display the report in multiple languages, then do the following:
 - a In the Parameters list, make sure the report parameter that Siebel CRM must translate is chosen.
 - b In the Parameter Translation list, click New.
 - c Define the following fields:
 - **Default Value.** Enter the default value that Siebel CRM displays in the Run Report pane for the language that you specify in the Language field.

- **Display Name.** Enter the label that Siebel CRM displays for this report parameter in the Parameter section of the Run Report pane for the language that you specify in the Language field.
 - **Language.** Choose the language that Siebel CRM must display in the report that it runs.
 - d Repeat [Step a](#) through [Step c](#) for each language that Siebel CRM must translate for this report.
- 8 Click Validate to determine if an inconsistency exists between the report template you defined in [Step 4](#) through [Step 7](#) and the report template.

Siebel CRM does the following work:

- Makes sure the number of parameters that Siebel CRM displays in the Run Report pane equals the number of parameters defined in the report layout template. Siebel CRM maps parameter types to the Report Parameter Form Applet business component field, except for Label parameter types. It ignores Label parameter types during validation.
 - Makes sure the Name attribute that Siebel CRM displays for each report parameter in the Run Report pane includes an equivalent placeholder in the report layout template.
- 9 (Optional) If you add a large number of report parameters, then add a scrollbar to the Run Report pane.

For more information, see [“Adding Scrollbars to the Parameters Section of the Run Report Pane” on page 112.](#)

Adding Report Parameters in Oracle BI Publisher Desktop

This topic describes how to add report parameters in Oracle BI Publisher Desktop.

To add report parameters in Oracle BI Publisher Desktop

- 1 Use Oracle BI Publisher Desktop to open a report template.
- 2 Use the following syntax to add a report parameter:

```
<?param@begin: Param1>
```

For example, you can add the following parameters:

```
Param1 = <?$Param1?>  
Param2 = <?$Param2?>  
Param3 = <?$Param3?>
```

For example, you can set Param1 to Active so that if the user chooses Param1, then the report only includes service requests that are active.

You can use a field that the user specifies to display data rows in different colors. For example, the ACTIVE_STATUS field.

For important caution information, see [“Configuring Report Parameters to Filter Data” on page 111.](#)

- 3 Make sure that a definition exists in the Siebel application for the parameter you added in [Step 2](#).

You must make sure that a parameter definition exists in the Siebel application for each report parameter that you specify, and that the Name property of this definition uses the same value that you specify in Oracle BI Publisher Desktop, such as Param1. For more information about parameter syntax and usage, see the topic about defining parameters in the *Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

- 4 (Optional) Configure Siebel CRM to display the parameter settings that the user provides.

To verify that Oracle BI Publisher receives the correct input, you can configure Siebel CRM to display the parameter settings that the user provides. Siebel CRM can display this information in the report output. This verification does not affect the report layout controls. For example, assume you use the following syntax to define param1 in the report template:

```
<?param@begin: Param1?>
```

You then add the following code to display the parameter values that the user specifies:

```
<?${Param1?}>
```

Configuring Report Parameters to Filter Data

CAUTION: It is recommended that you do not configure report parameters to filter data in a report template. Doing this can degrade performance.

It is recommended that you do not configure a report parameter that filter records. Instead, it is strongly recommended that the user run a query that filters records in the client. If you configure a report parameter that filters records, then Oracle BI Publisher will examine each record to determine if it matches the filter that you specify, and then filter these records. It will examine the entire record set again according to the parameters that the user specifies in the client. This configuration might degrade performance. For more information, see [“Caution About Running Reports with a Large Number of Records”](#) on page 22.

Report Parameter Types You Can Specify

[Table 10](#) describes the report parameter types that you can specify in the Type field of the Parameters list.

Table 10. Report Parameter Types

Parameter Type	Description
Check Box	Displays a check box.
Date	Displays a text box where the user can enter a date value and use a calendar control. The date format uses the format that the user preferences specify.

Table 10. Report Parameter Types

Parameter Type	Description
LOV	<p>Displays a picklist that allows the user to choose a value. You can define any list of values that references the Picklist Generic business component. An object definition must exist for the list of values that you reference. The LOV parameter type supports an MLOV (multilingual list of values). It does not support a dynamic or a hierarchical list of values.</p> <p>If the picklist is:</p> <ul style="list-style-type: none"> ■ Bounded. The LOV parameter is bounded. ■ Not bounded. The LOV parameter is not bounded.
Text	Displays a text box that allows the user to enter a string.
Number	Displays a text box that allows the user to enter a number. It also displays a calculator icon that allows the user to open a calculator.
DateTime	Displays a text box that allows the user to enter a date and time. It also displays a calendar icon that allows the user to open a calendar.
Label	<p>Displays text that you can use to describe part of the Parameters section in the Run Report pane. No control type is associated with this parameter type.</p> <p>You can use the Label parameter type only with the Order field. Siebel CRM aligns a Label parameter type along the edge (left) of the Run Report pane. It uses the longest label that you define as the starting position. It aligns all other parameter types relative to this starting position.</p> <p>You cannot use the Label parameter type to specify font face, size, color, style, alignment, or wrap in HTML.</p>
TextArea	Displays a text area that includes a scrollbar that allows the user to enter a large amount of text.

Adding Scrollbars to the Parameters Section of the Run Report Pane

Siebel CRM displays report parameters in the Parameters section of the Run Report pane in a single list. If you specify a large number of report parameters in a report template, then the user might not be able to choose some values, particularly if the user uses a computer that is set to a small screen resolution. You can configure Siebel CRM to display a scrollbar to avoid this problem.

To add scrollbars to the Parameters Section of the Run Report Pane

- 1 Log in to Siebel Tools.
- 2 In the Object Explorer, click Applet.
- 3 In the Applets list, query the Name property for Report Parameters Applet.

- 4 In the Object Explorer, expand the Applet tree, and then click Applet User Prop.
- 5 In the Applet User Properties list, query the Name property for Display Scrollbar.
- 6 Modify the Value property, as necessary.

Siebel CRM displays the scrollbar if the report template includes the number of parameters that you specify in the Display Scrollbar property. For example, if you specify five report parameters in the report template, and if you specify a value of 5 in the Display Scrollbar property, then Siebel CRM displays a scrollbar. Siebel Tools sets the value of the Display Scrollbar property to 25, by default.

Siebel CRM uses this configuration for all reports that use report parameters and for all users.

- 1 Compile the Siebel runtime repository.

Guidelines for Defining Parameters in Report Templates

If you define report parameters in a report template, then apply the following guidelines:

- Siebel CRM sends each report parameter value that a user enters to the Oracle BI Publisher Server as a string. You must make sure that your configuration includes the functions that some data types require so that Siebel CRM can recognize the data type. For example, you can use the canonical date functions to convert a string value to a date value so that Siebel CRM can recognize a Date data type.
- The minimum and maximum validation (date values entered in a set range) and validation or defaults using date functions, such as Today()-7 and conditional value logic (a display of one parameter that depends on another value) are not currently possible.
- The Report Parameters virtual business component includes a predefined number of parameter fields, by default. You can create more fields in this business component to add more report parameters for each data type. The following table describes the number of parameter fields that Siebel CRM provides, by default.

Data Type	Number of Parameter Fields
LOV	10
Text	5
Text Area	5
Number	10
Date	5
Date/Time	5
Check box	10

Customizing Other Options for Siebel Reports

This topic describes other options that you customize for Siebel Reports. It includes the following information:

[“Modifying the Sequence of Report Names in the Run Report Pane” on page 114](#)

Modifying the Sequence of Report Names in the Run Report Pane

This topic describes how to specify the sequence that Siebel CRM uses to display reports in the Report Name list in the Run Report pane. The Sequence field references a business component field. You can modify this field to customize how Siebel CRM displays the report. For more information about configuring business components, see *Configuring Siebel Business Applications*.

To modify the sequence of report names in the Run Report pane

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - BIP Publisher Reports screen, and then the View Association view.
- 3 Choose the view where you must define the report order.
- 4 In the Views list, choose the view you must modify.

For information about how Siebel CRM displays reports in the Report Name list, see [“How Siebel CRM Displays Reports According to the View” on page 14](#).

- 5 In the Report List, enter one of the following values in the Sequence field of each report that Siebel CRM displays in this list:
 - **0 (zero)**. Siebel CRM does not display this report in the Report Name list.
 - **No value or NULL**. Siebel CRM sequences the reports according to the report name in ascending alphabetic order according to the first letter that the report name contains. Siebel CRM uses this sequence as the default sequence.
 - **A positive integer**. Siebel CRM displays the reports in numeric ascending order first, and then displays reports that include no value or a NULL value.
 - **The same value for two or more reports**. Siebel CRM displays the reports in ascending alphabetic order according to the first letter that the report name contains.

8

Customizing Master-Detail Reports

This chapter describes how to customize a master-detail report. It includes the following topics:

- [About Master-Detail Reports](#)
- [Viewing the Report Template of a Master-Detail Report](#)
- [Creating a Master-Detail Report Template in Microsoft Word](#)

About Master-Detail Reports

A *master-detail report* is a type of report that displays a record that resides in a master business component and a list of the detail business component records that reference the master business component. The master business component possesses a one-to-many relationship with the detail business component. It is similar to a master-detail view in a Siebel application, in that Siebel CRM displays detail records for each master record. A master-detail view displays the detail records for one master record at a time, but a master-detail report displays detail records for all master records at the same time.

Figure 10 includes the Service Request Activity - All report in Siebel Service, which is an example of a master-detail report.

The screenshot shows a Siebel report titled "Service Request Activity (All)". The report header includes the Siebel logo, the title, and the report date: "Report Date: 29-OCT-2008 06:43:06 Page 1 of 6".

SR No. 83602-501161		Account Name Abbey General Cardiology		Status	Open
Date Opened	5/31/02 10:40 PM	Severity	4-Low	Description	
Date Closed		Priority	3-Medium		
Customer Reference Id		Owner	SMASTERS		
Activities					
Date Opened	Created By	Assigned To	Activity	Status	Description
	MDADMIN	MDADMIN	Field Repair	In Progress	Field level replacement
	MDADMIN	MDADMIN	Test Drive	Done	Stress test and certify unit
	MDADMIN	MDADMIN	Diagnosis	Done	Field-level diagnosis

Figure 10. Example of a Master-Detail Service Request Activity (All) Report

In this example, the master-detail reports provide master information for each service request, and a list of activities for this service request. Each service request begins on its own page. For more information, see [“Viewing the Report Template of a Master-Detail Report” on page 116](#).

A master-detail report can also include multiple detail elements where a list of detail records can display several business components for each master record. For example, the Account Service Profile report includes the following lists for each account master record:

- Customer survey responses
- Opportunities

■ Service requests

For a description of a report that includes two detail elements, see [“Creating a Master-Detail Report Template in Microsoft Word” on page 120](#).

Many reports that come predefined with Siebel CRM are master-detail reports.

How Siebel CRM Handles Multivalue Fields

The business object that the integration object references specifies how the master-detail report incorporates a parent business component and child and grandchild business components. When Siebel CRM runs a report, it captures the user interface context, and then sends it to the primary integration component of the integration object. If a report includes a multivalue field (MVF), then Siebel CRM displays only the first record. To display all the records from a multivalue field, you must create an integration object that references the business component that contains the multivalue field that Siebel CRM must display. For more information, see [“How Siebel Reports Uses Integration Objects” on page 82](#).

Siebel CRM also uses this configuration for an indirect multivalue field. For example, consider the case where the business address of an account is associated with an opportunity that Siebel CRM displays in the report. The business addresses in the multivalue field are not directly related to the opportunity, but they are related to the account that it is associated with this opportunity. To display all the records in the business address multivalue field as a detail section, you must do the following work:

- 1 Create a link between the Business Address business component and the Opportunity business component using Account Id as the source field.
- 2 Include the Business Address business component under the Opportunity business object
- 3 Create an integration object with the Business Address business component under the integration object, and include the required multivalue field.

For more information about multivalue fields, see *Configuring Siebel Business Applications*.

Viewing the Report Template of a Master-Detail Report

In the example in this topic, you view the report template that Siebel CRM uses for the Service Request Activity (All) master-detail report.

To view the report template of a master-detail report

- 1 Run the Service Request Activity (All) report:
 - a Log in to the Siebel Service application.
 - b Navigate to the Service screen, and then choose All Service Requests across Organizations.
 - c In the All Service Requests across Organizations list, choose an account and then click the Reports button in the application toolbar.

- d In the Run Report pane, in the Report Name list, choose Service Request Activity (All).
- e Choose a report output type, and then click Submit.
- f In the File Download dialog box, choose Open.

Siebel CRM displays the Service Request Activity (All) report in a browser window. To view this report, see [Figure 10 on page 115](#).

- 2 Open Oracle BI Publisher Desktop.

For more information, see [“About Oracle BI Publisher Desktop” on page 15](#).

- 3 Open the srvreqaa.rtf file that resides in the following folder:

SIEBEL_SERVER_ROOT\XMLP\TEMPLATES

For more information, see [“Directory Structure That Siebel Reports Uses” on page 86](#)

The following screen capture includes some of the features that this template uses:

The screenshot shows a Siebel report template for "Service Request Activity (All)". The report header includes the Siebel logo, the report title, and the report date. The main body of the report is a table with columns for SR No., Date Opened, Date Closed, Customer Reference Id, Account Name, Severity, Priority, Owner, Status, and Description. The table is generated using a series of SQL and Siebel-specific tags, including <code>for-each-group</code>, <code>for-each</code>, and <code>end</code>.

For example, this template includes the following items:

- Table for the master record that Siebel CRM displays in a form.
- Table for each set of child records that Siebel CRM displays in a list.
- For-each loop that Siebel CRM runs for all master records. It includes the parent form and all child lists.
- The following condition for the Activity section that prevents Siebel CRM from displaying header information in the Activity section if no activities exist for the service request:

`<?i f: current-group()//ssAction?>`

For information about how to identify the report template that is associated with a report, see [“Views You Use to Register Report Templates” on page 94](#).

- In the Oracle BI Publisher menu, choose Tools, Field Browser, and then Show All.

The Field Browser dialog box displays, that allows you to view and modify the template. For more information, see “Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template” on page 118.

Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Table 12 describes the code that Oracle BI Publisher uses for the Service Request Activity (All) report template. You can view this code if you use Oracle BI Publisher Desktop to open the Service Request Activity (All) report template. For more information about this code, see the *Fusion Middleware Report Designer’s Guide for Oracle Business Intelligence Publisher* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>). For more information about integration objects, see “How Siebel Reports Uses Integration Objects” on page 82.

Table 12. Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Syntax	Oracle BI Publisher Code	Description
If	<?i f: //ssServi ceRequest?>	An IF condition that determines if service request record exist, do not exist, or are not displayed.
for-eachgroup: ssServiceRequest	<?for-each- group: ssServi ceRequest; posi ti on()?>	The start of a group section that iterates for each record in the Service Request data set.
ssSrNumber ssAccount ssStatus	<?ssSrNumber?> <?ssAccount?> <?ssStatus?>	Field mappings.
If	<?i f: //ssCreated?>	If condition that does not display the ssCreated field if null.
ssCreated	<?format- date: psfn: totext (ssCreated, "yyyy-MM-dd' T' HH: mm: ss", "MM/dd/yyyy hh: mm: ss"); ' SHORT_ TI ME' ?>	Field formatting.
End	<?end i f?>	End of if condition.
ssSeverity ssDescription	<?ssSeveri ty?> <?ssDescri pti on?>	Field mappings.
If	<?i f: //ssCl osedDate?>	If condition that does not display the ssClosedDate field if null.
ssClosedDate	<?format- date: psfn: totext (ssCl osedDa te, "yyyy-MM-dd' T' HH: mm: ss", "MM/dd/yyyy hh: mm: ss"); ' SHORT_ TI ME' ?>	Field formatting.

Table 12. Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Syntax	Oracle BI Publisher Code	Description
End	<?end if?>	End of if condition.
ssPriority ssCustomerRefNumber ssOwner	<?ssPriority?> <?ssCustomerRefNumber?> <?ssOwner?>	Field mappings.
for-each:current-group()	<?for-each:current-group()?>	Start of a group section, iterating for each record in the current group (for example, Service Request).
If	<?if:current-group()//ssAction?>	An IF condition that determines if action records exist, do not exist, or are not displayed.
for-each:ssAction	<?for-each:ssAction?>	Beginning of the repeating element group for Action.
If	<?if://ssCreated?>	If condition that does not display the ssCreated field if null.
ssCreated	<?format-date:psfn:totext(ssCreated, "yyyy-MM-dd' T' HH:mm:ss", "MM/dd/yyyy hh:mm:ss"); 'SHORT_TIME' ?>	Field formatting.
End	<?end if?>	End of if condition.
ssCreatedByN ssOwnedBy sssStatusType ssDescription	<?ssCreatedByName?> <?ssOwnedBy?> <?ssType?> <?ssStatus?> <?ssDescription?>	Field mappings.
end for-each:ssAction	<?end for-each?>	End of the repeating element group for Action.
End if	<?end if?>	The end of the if condition
end for-each	<?end for-each?>	End of current-group() repeating group section.
page break	<?split-by-page-break: ?>	Page break after the end of the group section.
end for-each-group: ssServiceRequest	<?end for-each-group?>	End of Service Request group section.
End	<?end if?>	End statement that ends a subroutine.

Table 12. Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Syntax	Oracle BI Publisher Code	Description
If	<?i f: not (// ssServi ceRequest)?>	If condition that prints "No Data Found" in the report if there are no records.
End	<?end i f?>	End if statement for the condition in the first row.

Creating a Master-Detail Report Template in Microsoft Word

Creating a master-detail report follows the same process as creating a simple report. However, consider the following important points when creating master-detail reports:

- Make sure that all child integration components in the integration object include the required value in the Parent Integration Component property. For more information, see ["How Siebel Reports Uses Integration Objects"](#) on page 82.
- In the report template file, make sure that the correct for-each-group and for-each:current-group elements are placed in the template, see ["Viewing the Report Template of a Master-Detail Report"](#) on page 116.
- When creating custom integration objects for Siebel CRM, make sure that the fields used for dynamic queries in the applet of the master component are also contained in the integration component. *Dynamic queries* are specific, customized queries that you create.

To create a master-detail report template in Microsoft Word

- 1 Open Oracle BI Publisher Desktop in Microsoft Word, and then create a new document.
For more information, see ["About Oracle BI Publisher Desktop"](#) on page 15.
- 2 In the Oracle BI Publisher Desktop menu, choose Data, and then Load XML Data.
- 3 Choose the XML file that contains the sample data, and then save the template in RTF format.
- 4 Add titles, formatting, page header, page footer, and so on, as necessary.
- 5 Add the Master data section:
 - a Insert a master data section.
You can insert this section manually or use the Oracle BI Publisher Desktop menu, and then the Insert menu item.
 - b Choose the master data section, and then add a for-each-group section:
 - Click Insert, Repeating Group, and then the Advanced tab.
This process embeds the for loop in the table and does not replicate the location of the controls in the predefined examples.

- Change the expression:
`<?for-each-group: ssServiceRequest; position()?>`
- c (Optional) If you must view descriptive text for the inserted sections, click Tools, Options, Build, and then Descriptive.
- 6 Add the Detail data section:
 - a Choose the detail data section.
 - b In the Oracle BI Publisher menu, click Insert, and then Repeating Group.
 - c Choose the defaults, and then click OK.
- 7 Insert a page break after the final detail section:
 - a Choose the Oracle BI Publisher properties of the entire group.
 - b Navigate to Create, Group, and then Properties.
 - c Choose a page break option.
- 8 Add formatting features and conditions.
For more information about adding features, see [“Viewing the Report Template of a Master-Detail Report” on page 116](#).
- 9 Register the report.
For more information, see [“Registering Report Templates” on page 93](#).

Viewing Descriptive Text for Repeating Group Sections

When you create a template that includes a repeating group section, you might find it useful to view the descriptive text for the inserted sections. Descriptive text allows you to view the Oracle BI Publisher syntax.

To view the descriptive text for the repeating group sections

- 1 Open Oracle BI Publisher Desktop in Microsoft Word.
- 2 In the Oracle BI Publisher application-level menu, click Options, and then Build.
- 3 In the For-each form field box, choose Descriptive.

9

Deploying Reports

This chapter describes how to deploy reports from a development, test, or production environment to another environment. This chapter only describes how to deploy with a disconnected client. It includes the following topics:

- [Process of Deploying Reports on page 123](#)
- [Deploying Integration Objects to the Production Environment on page 128](#)

Process of Deploying Reports

You can use Application Deployment Manager (ADM) to deploy reports from a development or test environment to a production environment. You use ADM to prepare the report template data for deployment. You use it to consolidate report information into a single deployment package. A *deployment package* is a set of files that represent the report files. Siebel CRM stores these files in a predefined package folder. It also stores a package descriptor file that includes the details of the package contents. For more information about ADM, see *Siebel Application Deployment Manager Guide*.

To deploy reports, do the following tasks:

- 1 ["Preparing the Deployment Environment" on page 123](#)
- 2 ["Packaging the Report Files in the Development Environment" on page 124](#)
- 3 ["Deploying the Report Files to the Production Environment" on page 125](#)
- 4 ["Packaging Report Database Records in the Development Environment" on page 126](#)
- 5 ["Deploying Report Database Records to the Production Environment" on page 127](#)
- 6 (Conditional) ["Deploying Integration Objects to the Production Environment" on page 128](#)

Preparing the Deployment Environment

This task is a step in ["Process of Deploying Reports" on page 123](#).

To prepare the deployment environment

- 1 Make sure you finished developing and testing reports in the development environment.
- 2 Do the following work in the development environment:
 - a Install version 5.0 or later of the Perl software.

- b** Make sure you include the Application Deployment Manager (ADM) component group as part of the Siebel Server installation process.

For more information about installing Siebel Servers, see *Siebel Installation Guide* for the operating system you are using.

- c** Set up ADM so that it meets your deployment requirements. You must use a Siebel Management Server and you must install Siebel Management Agents.

For more information about setting up ADM, see *Siebel Installation Guide* for the operating system you are using.

Packaging the Report Files in the Development Environment

This topic describes how to package the report files in the development environment.

This task is a step in [“Process of Deploying Reports” on page 123](#).

To package the report files in the development environment

- 1** Manually copy the files that you must deploy from the `SI EBSRVR_ROOT\XMLP` folder in the development environment to the following folder in the production environment:

`SharedFolder\PackageName\file\siebsrvr\XMLP\subfolder_name`

where:

- `SharedFolder` is a shared folder that resides on a computer in the production environment.
- `PackageName` is the name of the ADM package folder that contains the report files that you must deploy.
- `file` is the type of file, such as database, repository, and so on.
- `siebsrvr\XMLP\subfolder_name` is the Siebel Server folder structure.

- a** Copy the XML files to the following folder:

`SharedFolder\PackageName\file\AppServer\XMLP\DATA`

- b** Copy the RTF and XSL files to the following folder:

`SharedFolder\PackageName\file\AppServer\XMLP\TEMPLATES`

- c** Copy the XLIFF files to the following folder:

`SharedFolder\PackageName\file\AppServer\XMLP\enu\xliff`

- 2** Run the following command to create a descriptor file for the package:

```
admpkgr generate "shared folder\package name"
```

where:

- `shared folder\package name` is the shared folder and package name where the packages that you must deploy reside.

Deploying the Report Files to the Production Environment

This topic describes how to deploy the report files to the production environment.

This task is a step in [“Process of Deploying Reports” on page 123](#).

To deploy the report files to the production environment

- 1 Navigate to the installation folder where the Siebel Management Server is installed.

- 2 Run the following command to create a package folder structure:

```
admpkgr i n i t SharedFolder\PackageName
```

where:

- `SharedFolder\PackageName` is the shared folder package name where the packages that you must deploy reside.

You must specify a path in a shared location.

If an error occurs, then the path to the JAR files might not be correct. To fix this error, make sure the CLASSPATH variable references the correct path to the JAR files in the `admpkgr.bat` file in the Management Server installation folder.

- 3 Run the following command to call the `deploy.bat` file to load the package to the server database:

```
depl oy_ enterprise l o a d username password PackageName
```

where:

- `enterprise` is the name of the Siebel enterprise that you provide during configuration.
- `username` and `password` are the username and password account of the Siebel user who is deploying the package.
- `PackageName` is the name of the package you created in [Step 2 on page 124](#).

- 4 Run the following command to create a session for the deployment:

```
depl oy_ enterprise create username password PackageName
```

- 5 Run the following copy command to deploy the package:

```
depl oy_ enterprise copy username password PackageName
```

- 6 Navigate to the following folder on the Siebel Server:

```
si ebsrvr\XMLP
```

- 7 Verify that the files you placed in the deployment package described in [Step 1 on page 124](#) are available in the correct folders.

Packaging Report Database Records in the Development Environment

This topic describes how to use Application Deployment Manager (ADM) to package report database records.

This task is a step in [“Process of Deploying Reports” on page 123](#).

To package report database records in the development environment

- 1 Navigate to the Application Deployment Manager screen, and then the Deployment Projects view.
- 2 In the Deployment Projects list, create a new project record.
- 3 Complete the project fields, as required.

Make sure the Export to File field contains a check mark so that Siebel CRM creates the export file. It is recommended that you include a BIP prefix in report project name. This prefix allows you to search for your report projects. For descriptions of the project fields, see *Siebel Application Deployment Manager Guide*.

- 4 In the Deployment Filter field in the Deployment Project data type list, create search expressions for the data types to filter only those items of a data type that match the condition for deployment. Save each filter.

For example, assume you create a report in the development environment, and this report includes the following items:

- Named *Test Account List*
- References the BIP Accounts - Current Query integration object
- Associated with the Account List view

In this example, you enter filter information to deploy the data types to the production environment using information from the following table.

Data Type	Filter
BIP Sample Data Generation	[Name]=' BIP Accounts - Current Query'
BIP Report Template Registration	[Report Name]=' Test Account List'
BIP View Association	[Name]=' Account List View'
BIP Report Template Translations	[Report Name]=' Test Account List'

If a data type is not available, then it might be set to Inactive. Make sure the Active field for each data type contains a check mark. The Active field resides in the Data Type Details view.

This step allows you to query the reports that you are deploying to the production environment. For more information about search expressions, see *Configuring Siebel Business Applications*.

- 5 Enable the report project:
 - a Navigate to the Deployment Projects view.

- b In the Deployment Projects list, choose the draft deployment.

The Status field of the draft deployment project record displays as Draft.

- c Click the Enable button to activate the report project.

ADM populates the Status field with Enabled and the Publication Date/Time field with the date and time of the report project activation.

6 Export the reports:

- a Navigate to the Application Deployment Manager screen, and then the Deployment Sessions view.
- b Choose the report project you must enabled.
- c Make sure the Export to File and the Deployment Lock fields each contain a check mark.
- d Enter a shared location to store the XML files that contain the database records.
- e Click Deploy.

ADM deploys the database record XML files to the shared location.

Deploying Report Database Records to the Production Environment

This topic describes how to deploy report database records from the development environment to the production environment.

To deploy report database records to the production environment

- 1 Navigate to the Application Deployment Manager screen, and then the Deployment Sessions view.
- 2 Click the Deployment Sessions Menu button, and then choose Deploy from File.
- 3 In the Deploy from File dialog box, enter the file paths from which to deploy the database records:
 - a Enter the file path for the XML file that contains the *Sample Data Generation* file, and then click Import.
 - b Enter the file path for the XML file that contains the *Report Template Registration* file, and then click Import.
 - c Enter the file path for the XML file that contains the *View Association* file, and then click Import.

Make sure you specify the shared location where the database records are stored.

For example, you might enter the following shared locations:

- \\sharedlocation\88-25ZC7_BIP_Sample_Data_Generation.xml
- \\sharedlocation\88-25ZC7_BIP_Report_Template_Registration.xml
- \\sharedlocation\88-25ZC7_BIP_View_Association.xml

- 4 In the production environment, make sure ADM deployed the data. Verify that Siebel CRM displays the report in the Run Report pane.

For example, in [Step 4 on page 126](#), you created a report named Test Account List. To verify that ADM deployed this report correctly, do the following:

- Navigate to the Accounts screen, and then the Account List view.
- Click the Reports button in the application toolbar.
- Verify that the Report Name list in the Run Report pane includes the *Test Account List* report.

Deploying Integration Objects to the Production Environment

If you modified a predefined integration object, or created a new one, then you must deploy it to the production environment. For more information, see ["Modifying Integration Objects" on page 100](#).

Use the Siebel Migration Application Incremental Runtime Repository Data Service to migrate and deploy integration object changes to your target environment.

For more information about Siebel Migration and the Migration Application Incremental Runtime Repository Data Service, see *Siebel Database Upgrade Guide*.

10 Troubleshooting Siebel Reports

This chapter describes how to troubleshoot Siebel Reports. It includes the following topics:

- [Enabling Logging for Siebel Reports in the Siebel Application on page 129](#)
- [Fixing Class Not Found Errors on page 132](#)
- [Fixing Class Not Found Errors When Previewing Reports in Microsoft Word on page 133](#)
- [Troubleshooting Error Messages for Siebel Reports on page 134](#)

Enabling and Disabling Logging for Siebel Reports

This topic describes how to enable and disable logging for Siebel Reports. It includes the following information:

- [“Enabling Logging for Siebel Reports in the Siebel Application” on page 129](#)
- [“Enabling Logging and Debugging on the Oracle BI Publisher Server” on page 131](#)
- [“Enabling and Disabling Debugging in Disconnected Clients” on page 131](#)

Enabling Logging for Siebel Reports in the Siebel Application

You can configure Siebel CRM to create log files that capture detailed information about errors that occur while running reports. The reports administrator can use the log files to investigate why the error occurred.

You set the log level for the XMLP Report server component by using the Server Manager UI or the Server Manager command-line interface program (srvrmgr program).

You can adjust the log levels at any time.

Oracle BI Publisher logs all exceptions and debug level information.

Setting the Log Level for the XMLP Report Server Component

This topic describes how to set the log level for the XMLP Report server component.

To set the log level for the XMLP Report server component

- 1 Log in to the Siebel client with administrator privileges.

- 2 Navigate to the Administration-Server Configuration screen, Servers, and then the Components view.
- 3 In the Component field, query for XMLP Report Server, and then click the Events tab.
- 4 In Events list, query for XMLP Report Log, and then set the value for the log level to 5.

NOTE: Setting logging levels to a high value for all components in report generation, such as the requesting Object Manager, the XMLP Report Server component, the EAI Object Manager, the File System Manager and the BI Publisher Server affects performance. The higher the logging-level value, the slower the performance.

- 5 Click the Component tab.
- 6 In the Component field, query for the Application Object Manager, and then click the Events tab.
For example, query for the following Application Object Manager:

Call Center Object Manager (ENU)

- 7 In the Events list, query for XMLP Report Log, and then set the value for the log level to 5.
- 8 Stop, and then restart the XMLP Report Server and Siebel Application Object Manager server components.

For more information about restarting the Siebel Server, see *Siebel System Administration Guide*.

Using the Server Manager Command-Line Interface to Set Log Levels for the XMLP Report Server Component

This topic describes how to use the Server Manager command-line interface to set the log level for the XMLP Report server component.

To use the Server Manager command-line interface to set log levels for the XMLP Report server component

- 1 Run the following command to connect to the Siebel Server Manager:

```
srvrmgr /g gateway computer name:port number /s xmlp siebel server name  
/e enterprise /u user name /p password
```

where:

- *gateway computer name* is the name of the computer where the Siebel Gateway.
- *port number* identifies the port number where the Siebel Gateway is listening.
The colon and the port number are optional if using an optional default port.
- *xmlp siebel server name* is the name of the server where the XMLP Report server component is enabled.
- *user name* is the login name of the administrator.
- *password* is the password for the administrator.

For example:

```
srvrmgr /g gateway computer name: 2330 /s xmlp siebel server name /e Siebel  
/u SADMIN /p MSSQL
```

- 2 Run the following command to change the event log level:

```
change evtloglvl XMLPReportLog=5 for comp xmlpreportserver
```

- where 5 is the event log level that you must change.

For example, to get details for a call to a business service, run the following command:

```
change evtloglvl ObjMgrBusServiceLog=5 for comp xmlpreportserver
```

- 3 Shut down, and then restart the XMLP Report server component.

Enabling Logging and Debugging on the Oracle BI Publisher Server

This topic describes how to enable logging and debugging on the Oracle BI Publisher Server.

Enabling Logging on the Oracle BI Publisher Server

You can configure the Oracle BI Publisher Server to create log files that collect detailed information about errors that occur when the user runs a report in a connected client. You can use these log files to determine how much time Siebel CRM requires to run a report, and so on. For information about enabling logging for the Oracle BI Publisher Server, see the *Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher* and *Fusion Middleware Developer's Guide for Oracle Business Intelligence Publisher* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>).

Enabling Debugging on the Oracle BI Publisher Server

You can configure Oracle BI Publisher Server to capture detailed information about errors that occur while working with Siebel Reports in the Siebel Web Client. This debugging is in addition to the typical debugging capabilities that are available in Siebel CRM. For information on enabling debugging on the Oracle BI Publisher Server, see 1628945.1 (Article ID) on My Oracle Support.

Enabling and Disabling Debugging in Disconnected Clients

You can configure the Oracle BI Publisher XDO Engine to create log files that collect detailed information about errors that occur when the user runs a report in a disconnected client. You can use the log files to identify problems that occur with the JAR (Java ARchive) files or the Oracle BI Publisher report template files.

Enabling Debugging in Disconnected Clients

Use the following procedure to enable debugging for the Oracle BI Publisher XDO Engine.

To enable debugging in disconnected clients

- 1 Use a text editor to create a file named `xdodebug.cfg`.
- 2 Add the following code to the file you created in [Step 1](#):

```
LogLevel =STATEMENT  
LogDir=path to logging folder
```

- 3 Place the file you created in [Step 1](#) in the `jr\lib` folder that the Siebel client uses.

For example:

```
C:\Program Files\Java\jre1.6.0_07\lib
```

- 4 Make sure the folder that the `LogDir` parameter specifies exists.

Siebel CRM saves the log files that it creates in this folder. If an error occurs when the user runs a report in a disconnected client, then Siebel CRM creates an `xdo.log` file and saves it in the directory that the `.cfg` file specifies. You can use this log file to troubleshoot the problem.

Disabling Debugging in Disconnected Clients

You can disable debugging for the Oracle BI Publisher XDO Engine to optimize Siebel Reports performance in a disconnected client.

To disable debugging in disconnected clients

- 1 Back up the `xdo.cfg` file:
 - a Access the computer where the Siebel Developer Web Client is installed.
 - b Copy the `xdo.cfg` that resides in the `jr\lib` installation folder to another computer.
- 2 Remove or rename the `xdo.cfg` that resides in the `jr\lib` folder.

Fixing Class Not Found Errors

This topic describes how to fix a class not found error when you preview a report in Microsoft Word.

Guidelines for Fixing Class Not Found Errors When Previewing Reports in Microsoft Word

You can use the following guidelines to fix a class not found error when you preview a report in Microsoft Word:

- Make sure no spaces or new line characters exist after the following string:

```
-Xbootclasspath/a:
```

For example:

```
set_JAVA_OPTIONS=-Xbootclasspath/a: C: \81DQSSIA\cli ent\cl asses...
```

- To examine the syntax, turn off word wrapping in the editor you are using so that the editor displays the following code on a single line:

```
set_JAVA_OPTIONS
```

If you turn off word wrapping, then you must include three separate lines in your batch file.

- Make sure the path to Microsoft Word that the batch file references is correct. You can search for winword.exe from C: \Program Files to locate the path.
- The %1 in the batch file is an argument that represents a document that you typically open in Microsoft Word (RTF files).
- The following system environment variable prevents you from previewing reports in the Siebel client. Do not use it:

```
set _JAVA_OPTIONS
```

Fixing Class Not Found Errors When Previewing Reports in Microsoft Word

This topic describes how to fix an error that might occur when you preview a report in Microsoft Word that uses a predefined report template file. To fix this error, you create a custom batch file that loads the required template libraries before Microsoft Word opens the report template file. You replace the explicit drive and folder locations in this batch file with the drive and locations that your computer uses according to where you installed the Siebel Web Client. You are not required to associate a .doc extension with this batch file.

To fix class not found errors when previewing reports in Microsoft Word

- 1 Navigate to the following directory on the computer where you installed the Siebel client:

```
C: \Program Files
```

- 2 Create a new file named MSWordForBIPub.bat.
- 3 Add the following code to the file you created in [Step 2](#):

```
echo %1
```

```
set _JAVA_OPTIONS=-Xbootclasspath/a:
```

```
C: \81DQSSIA\cli ent\cl asses\Siebel XMLP. jar; C: \81DQSSIA\cli ent\cl asses\si ebel . jar; C: \81DQSSIA\cli ent\cl asses\XSLFuncti ons. jar; C: \81DQSSIA\cli ent\cl asses\Siebel CustomXMLP. jar; C: \81DQSSIA\cli ent\cl asses\Siebel CustomXMLP_SIA. jar
```

```
"C:\Program Files\microsoft office\Office\Word.exe" %1
```

This code sets the `_JAVA_OPTIONS` environment variable and then opens Microsoft Word.

- 4 Create a shortcut to the `MSWordForBIPub.bat` batch file and move it to your desktop.
- 5 If you must make sure this batch file always runs when you open a predefined report template file, then associate the RTF files with the `.bat` file:

- a Navigate to the folder that contains the template files in your Siebel client environment.

For example, navigate to the following folder:

```
C:\Siebel\client\XMLP\TEMPLATES
```

- b Sort the templates files according to Type.
- c Right-click on an RTF file, choose `Open With`, and then choose a text editor.
- d Click `Browse` to find the `.bat` file, and then open it.
- e Make sure the following check box contains a check mark, and then click `OK`:

Always Use the Selected Program to Open This Kind of File

This step associates the RTF file with the batch file you created in [Step 2](#). The batch file sets an environment variable that Oracle BI Publisher requires to publish a report in Microsoft Word.

If you double-click the RTF file, then a command prompt window displays. You can also use the desktop shortcut and run the shortcut to the `.bat` file to test the report preview.

Some RTF files in the following folder are set to read only:

```
C:\siebel\client\XMLP\TEMPLATES
```

To avoid a run-time error, you can right-click the template file, and then remove the check mark from the `Read-only` attribute.

Troubleshooting Error Messages for Siebel Reports

This topic describes guidelines for fixing error messages that occur with Siebel Reports. To fix one of these problems, look for it in the Symptom or Error Message column in [Table 13](#).

Table 13. Problems with Siebel Reports

Symptom or Error Message	Solution
SBL-OMS-00203: Error <?> invoking method "<?>" for Business Service "<?>"	<p>This error might occur if you fail to upload the template to the Oracle BI Publisher Server. It might occur in a connected or disconnected client.</p> <p>To fix this problem, see "Fixing Errors That Occur When Running Siebel Reports (SBL-OMS-00203)" on page 139.</p>

Table 13. Problems with Siebel Reports

Symptom or Error Message	Solution
<p>Siebel CRM displays the following message if you click Submit in the Run Report pane"</p> <p>"Unable to find definition for component XMLPReportServer(SBL-SRQ-00103)"</p>	<p>This error might occur after you click Submit to run a report. It might occur in a connected or disconnected client.</p> <p>To fix this problem, see "Fixing Errors That Occur After Clicking Submit in the Run Report Pane (SBL-SRQ-00103)" on page 139.</p>
<p>The specialized method 'GetFileToDir' is not supported on Business Component 'Report Template BC' used by Business Object 'Report Administration'. (SBL-DAT-00322)</p>	<p>This error might occur if the path settings in the configuration files are not correct. It might occur in a connected or disconnected client.</p> <p>To fix this problem, see "Fixing Errors That Occur Due to Incorrect Configuration File Settings (SBL-DAT-00322)" on page 139.</p>
<p>SBL-RPT-50504: Error occurred while saving the record. No data file provided for XLIFF generation.</p>	<p>This error might occur if you do not provide a sample XML data file for creating an XLIFF file. It occurs only in a disconnected client.</p> <p>To fix this problem, see "Fixing XLIFF File Errors" on page 141.</p>
<p>Siebel CRM displays the following message if you run a report:</p> <p>Class name incorrect</p>	<p>This error might occur if Oracle BI Publisher or the JAR files are not loaded. It occurs only in a disconnected client.</p> <p>To fix this problem, you set the classpath. For more information, see "Fixing the CLASSPATH Settings" on page 141.</p>
<p>SBL-EAI-04116 BIP: Server connection is lost</p>	<p>This error might occur with a very large or complex report. It occurs only in a connected client.</p> <p>You can increase the HTTPSleepTime method argument to fix this problem. For more information, see "Modifying the HTTP Sleep Time" on page 68.</p>
<p>Siebel CRM displays one of the following messages in the XMLP log file:</p> <ul style="list-style-type: none"> ■ Verify the BIP Server Userid and Password. (SBL-RPT-50529) ■ Failed to log into BI Publisher: invalid username or password. 	<p>This error message displays if you do not provide the correct user ID and password for the Oracle BI Publisher Server in the XMLP Report server component parameters. This problem occurs only in a connected client.</p> <p>To fix this problem, see "Fixing Oracle BI Publisher Server Login Errors" on page 141.</p>

Table 13. Problems with Siebel Reports

Symptom or Error Message	Solution
<p>Generic Error in Compression Routine (SBL-UIF-00227)</p>	<p>Make sure the temp\xml p folder exists in the <i>SI EBSRVR_ROOT</i> folder. If it does not, then add it. For more information, see “Directory Structure That Siebel Reports Uses” on page 86.</p> <p>Also, make sure the Siebel File System is set correctly. For information about setting the Siebel File System, see <i>Siebel System Administration Guide</i>.</p> <p>This problem might occur in a connected or disconnected client.</p>
<p>PublicReportService::executeCreate Report Failure: due to Report Name contains special characters for report</p>	<p>Make sure the report name does not contain the following special characters:</p> <ul style="list-style-type: none"> ■ Forward slash (/) ■ Backslash (\) ■ Double quote (") ■ Single quote (') ■ Tilde (~) ■ Ampersand (&) ■ Asterisk (*) ■ Plus sign (+) ■ Left angle bracket (<) ■ Right angle bracket (>) ■ Percent sign (%) <p>This problem occurs only in a connected client.</p>

Table 13. Problems with Siebel Reports

Symptom or Error Message	Solution
<p>SBL-EAI-04308: Operation 'runReport' of Web Service 'http://xmlns.oracle.com/oxp/service/PublicReportService.PublicReportServiceService' at port 'PublicReportService' failed with the following explanation: "oracle.apps.xdo.webservice.exception.OperationFailedException: PublicReportService::generateReport failed: due to oracle.apps.xdo.servlet.CreateException: Report definition not found: /SiebelCRMReports/<Report Name>/<Report Name>.xdo".</p>	<p>This error might occur if you run a report that includes missing information on the Oracle BI Publisher Server. A missing .xdo file is an example of missing information.</p> <p>To fix this problem, upload the report template from Oracle BI Publisher. For more information about uploading report templates, see "Registering Report Templates" on page 93.</p> <p>This problem occurs only in a connected client.</p>
<p>■ An column alignment problem occurs in HTML report output.</p>	<p>This problem might occur if the table header row and the data row reside in two different tables, but Siebel CRM displays them as a single table because no space exists between these tables. HTML or EXCEL might create this output.</p> <p>This problem might occur in a connected or disconnected client.</p> <p>To fix this problem, see "Fixing Alignment Problems in HTML" on page 142.</p>
<p>One of the following problems occur:</p> <ul style="list-style-type: none"> ■ An alignment problem occurs in report that runs in Japanese or PSJ. ■ The PPT output type for Japanese (JPN) is distorted in a report. 	<p>To fix this problem, you can create the font mappings for the Oracle BI Publisher Server to make sure that the HTML output displays correctly. This problem occurs only in a connected client.</p> <p>To fix this problem, see "Fixing Japanese Font Errors" on page 142.</p>

Table 13. Problems with Siebel Reports

Symptom or Error Message	Solution
<p>Siebel CRM displays one of the following message when you upload files to Oracle BI Publisher:</p> <ul style="list-style-type: none"> ■ SBL-EAI -05010: Class name incorrect or does not extend Siebel BusinessService : com/siebel/data/SiebelPropertySet -- JVM Exception: java.lang.NoClassDefFoundError: com/siebel/data/SiebelPropertySetObjMgrBusServiceLog ■ Object manager error: ([0] Class name incorrect or does not extend Siebel BusinessService : <?> 	<p>To fix this problem, verify the following:</p> <ul style="list-style-type: none"> ■ JAVA_HOME variable is set correctly. ■ The following JAR files are copied from the <i>siebel root</i>\classes\original folder to the <i>ORACLE_HOME</i>\j2ee\home\applications\xmlpserver\xmlpserve\WEB-INF\lib folder: <ul style="list-style-type: none"> ■ Siebel.JAR ■ SiebelXMLP.JAR ■ XSLFunctions.JAR ■ SiebelCustomXMLP.JAR ■ SiebelCustomXMLP_SIA.JAR ■ The CLASSPATH variable is set correctly in the JVMSubSys profile. ■ The JVM SubSystem Name parameter for XMLPReportServer is set to XMLPJvmSubsys. If this value is JAVA, then set it to XMLPJvmSubsys.
<p>Siebel CRM displays the following message when you upload files to Oracle BI Publisher:</p> <p>SBL-EAI -04308: Operation 'uploadReport' of Web Service 'http://xmlns.oracle.com/oxp/service/v11/PublicReportService.PublicReportServiceService' at port 'PublicReportService_v11' failed with the following explanation: "oracle.apps.xdo.webservice.exception.InvalidParametersException: PublicReportService: execute UploadReport Failure: Due to Report with Path [/Siebel CRMReports/Application Activity/Application Activity.xdo] already exist! ".</p>	<p>This error indicates that the Oracle BI Publisher Server already includes a copy of the file that you are uploading. The uploadReport method does not support uploading a new version of an existing file.</p> <p>To fix this problem, you must rename or delete the folder for the existing instance of the report on the Oracle BI Publisher Server. The Oracle BI Publisher Server stores the report template files in the following directory:</p> <p><i>ORACLE_HOME</i>\xmlp\XMLP\Reports\ Siebel CRMReports</p>

Fixing Errors That Occur When Running Siebel Reports (SBL-OMS-00203)

This topic describes how to fix an error that occurs while running Siebel Reports.

To fix errors that occur when running Siebel Reports

- 1 Make sure the XMLP Report server component is enabled.
- 2 Increase the XMLP Report server component log level to 5 to create a more detailed log file.
- 3 Copy the `xdodebug.log` file to the `jr\re\lib` folder.

For information about doing these tasks, see [“Enabling Logging for Siebel Reports in the Siebel Application” on page 129](#) and [“Enabling and Disabling Debugging in Disconnected Clients” on page 131](#).

Regenerating XML and Running the Report

If you create a report template with a database, but then register it with a different database, then the report might fail. This topic describes how to fix this problem.

To regenerate XML and run the report

- 1 Regenerate the XML data.
- 2 Reregister the report template.
For more information, see [“Registering Report Templates” on page 93](#).
- 3 Rerun the report.

Fixing Errors That Occur After Clicking Submit in the Run Report Pane (SBL-SRQ-00103)

This topic describes how to fix an error that occurs after you click Submit in the Run Report pane.

To fix errors that occur after clicking Submit in the Run Report pane

- 1 Enable and synchronize the XMLP Report server component.
For information about enabling this component, see *Siebel System Administration Guide*.
- 2 Restart the Siebel Server.
For information about restarting the Siebel Server, see *Siebel System Administration Guide*.

Fixing Errors That Occur Due to Incorrect Configuration File Settings (SBL-DAT-00322)

This topic describes how to fix an error that occurs due to incorrect configuration file settings.

To fix errors that occur due to incorrect configuration file settings

- 1 Make sure Siebel File System path is correct:
 - a Navigate to the Administration - Server Configuration screen, Servers, and then the Components view.
 - b Query for the object manager that your Siebel application uses.
For example, Call Center, Siebel Sales, or Siebel Service.
 - c Click the Parameters tab, query for FileSystemManager, and then click Advanced.
 - d In Siebel File System field, enter the following path:
`\\computer name\fs`
 - e Make sure that the fs folder exists in the D drive and that it is shared.
- 2 Restart the Siebel Server.
- 3 Verify that the File System component is running.
- 4 Verify that CLASSPATH is set in the JVMSubSys profile parameter.
- 5 Verify that all the required JAR files are included and available in the CLASSPATH.
- 6 Verify that the CLASSPATH separator is in the correct format for Windows or UNIX.
In Windows, the CLASSPATH separator is a semicolon (;). In UNIX, the CLASSPATH separator is a colon (:).

Fixing Errors in Disconnected Clients

This topic describes how to fix an error that occurs in a disconnected client.

To fix errors in disconnected clients

- 1 In the application .cfg file, set the following value for the FileSystem parameter:
`FileSystem \\computer name\fs\att`
- 2 Make sure the fs folder in the path that you added in [Step 1](#) is shared.
- 3 In the XMLPReports section of the .cfg file, make sure the following section is defined:

```
[XMLPReports]
XdoDir = /xml p/templates/
ReportOutputDir = /xml p/reports/
ReportDataDir = /xml p/data/
```
- 4 To determine if the path to the file system is set properly, try to register the report template.
If you can successfully register the report template, then you can drill down on the XLIFF file. If you cannot browse the report template, or if you cannot drill down on the XLIFF file, then you must make sure the path to the file system is set correctly.

Fixing XLIFF File Errors

This topic describes how to fix an error that occurs if you do not provide a sample XML data file when you create the XLIFF files.

To fix XLIFF errors

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - BI Reports screen, and then the Sample Data File Generation view.
For more information about this view, see [“Creating XML Files from Integration Objects” on page 103](#).
- 3 Choose the integration object associated with the report, and then click Generate Sample XML.

Fixing the CLASSPATH Settings

This topic describes how to fix the CLASSPATH parameter.

To fix the CLASSPATH parameter

- 1 Log in to the Server Manager command-line interface.
For information about how to use the `svrmgr` program, see *Siebel System Administration Guide*.
- 2 Run the following command to change the CLASSPATH:
Change param CLASSPATH =
C: \sba81\si ebsrvr\CLASSES\Siebel.jar; C: \sba81\si ebsrvr\CLASSES\Siebel XMLP.jar; C: \sba81\si ebsrvr\classes\xdo-core.jar for named subsystem XMLPJvmSubsys

Fixing Oracle BI Publisher Server Login Errors

This topic describes how to fix Oracle BI Publisher Server login errors.

To fix Oracle BI Publisher Server login errors

- 1 Log in to the Siebel client with administrator privileges.
- 2 Navigate to the Administration - Server Configuration screen, and then the Servers view.
- 3 Click the Component tab, and then query for XMLP Report Server.
- 4 Click the Parameters tab, query for BIP Server Login, and then enter the correct login information for the Oracle BI Publisher Server.
- 5 Query for the BIP Server Password, and then enter the correct password for the Oracle BI Publisher Server.

Fixing Alignment Problems in HTML

This topic describes how to fix alignment problems that occur in the HTML output of a report.

To fixing alignment problems in HTML

- 1 Add a row to the Table Header.
- 2 Copy the Form fields from the data row in the added row.
- 3 Delete the previous table that contained the data row.

The single table now contains header and data row.

Fixing Japanese Font Errors

Fixing Japanese Font Errors that occur with Oracle's Siebel Reports.

To fix Japanese font errors

- 1 Locate the msgothic.ttc file.
- 2 Copy the file you located in [Step 1](#) to the following folders on the computer where you installed the Oracle BI Publisher Server:
 - \$JAVA_HOME\jre\lib\fonts
 - ORACLE_HOME\common\fonts
- 3 Log in to Oracle's BI Publisher Server.
- 4 Navigate to Admin, Runtime configuration, and then Font Mappings.
- 5 Create the following font mappings:
 - Arial, Normal, msgothic.ttc
 - Arial, Bold, msgothic.ttc

Index

A

administering reports 61

C

copying JAR files to the Oracle BI Publisher Server 39

custom reports

creating new integration objects for 102

creating, process of 90

extending integration objects for Siebel Reports 101

process of creating, diagram of 90

D

debugging

disabling for Siebel Reports in disconnected mode 132

enabling for Siebel Reports in disconnected mode 132

enabling on Oracle BI Publisher for Siebel Reports 131

deleting scheduled reports 29

deploying reports to the Siebel Web Client from a disconnected client 95

directory structure for Siebel Reports

DATA directory, described 86

FONTS directory, described 86

REPORTS, directory described 86

TEMPLATES directory, described 86

xliff directory, described 86

documentation, where to find previous versions of Siebel Reports 18

E

enabling and configuring report scheduling, process of 57

enabling external file references for the Oracle BI Publisher Server 40

error messages, troubleshooting for reports 134

Extensible Markup Language

See XML

Extensible Stylesheet Language

See XSL

Extensible Stylesheet Language Transformation

See XSLT

Extensible Stylesheet Language-Formatting Objects

See XSL-FO

G

Generate Report pane

for Siebel Open UI, sample diagram of 13

generating reports

for selected records, about 22

in Siebel Business Applications, about 19

monitoring the status of 22

using the Siebel high-interactivity client, about 14

using the Siebel Open UI client 20

using the Siebel Open UI client, about 13

using the Siebel Open UI client, scenario for 20

workflow for connected mode 79

workflow for generating in disconnected mode 81

I

integrating Oracle BI Publisher with Siebel Business Applications

guidelines for 78

preparation tasks for 35

integrating Oracle BI Publisher with Siebel CRM

guidelines for 31

process of installing and configuring Oracle BI Publisher 36

roadmap for, diagram of 34

integrating Oracle BI Publisher with Siebel CRM for First-Time Installations

roadmap for 33

integration objects for Siebel Reports

creating new 102

extending to add new fields to a report 101

reducing the size of, about 104

setting search and sort order 21

sorting on date fields 21

working with multiple repositories 98

J

JAR files for Oracle BI Publisher Server,

adding an explicit reference to 54

L

locales

adding to report layout templates 73
deleting from report layout templates 73

logging for Siebel Reports

enabling logging for in the Siebel application 129
setting log levels for using the GUI 129
setting log levels for using the Server Manager (srvrmgr) 130

logging for the Oracle BI Publisher Server

about enabling 131

M

master-detail reports

about 115
creating master-detail layout templates in Microsoft Word 120
example of comparing the Service Request Activity (All) report with its corresponding layout template 116
using 115

migrating reports

migrating an integration object for a report 128
migrating new reports after creating new integration objects 128
migrating the report files to the target environment 125
moving database records to the target environment 127
packaging and migrating reports, process of 123
packaging database records in the source environment 126
packaging the report files in the source environment 124
requirements for 123

multilingual reports

about working with 71
adding and deleting locales from report layout templates 73
adding languages other than English 73
designing 73
uploading to the Siebel application, roadmap for 73

O

optimization

enabling scalable mode for Siebel Reports 69
flagging minimal fields as Force Active 104

increasing EAI HTTP transport sleep time for 68

optimizing performance for large data volumes, process of 67

selecting correct logging levels 130

setting a temporary directory on the Oracle BI Publisher Server for 70

setting concurrency parameters for 66

setting the report execution wait time for 65

setting the server request processor wait time for 66

Oracle BI Publisher

enabling debugging on Oracle WebLogic Server for Siebel Reports 131

starting and stopping 40

upgrading to the latest patch 41

Oracle BI Publisher Add-in for Microsoft Word

See Oracle Business Intelligence Publisher Add-in for Microsoft Word

Oracle BI Publisher Enterprise, installing 37

Oracle BI Publisher Publisher

creating reports 14

Oracle BI Publisher repository

about 80

Oracle BI Publisher Server

about 80

about installing 14

adding an explicit reference to JAR files 54

copying JAR files to 39

enabling external file references for 40

increasing the Java heap size for, about 70

Oracle BI Publisher-Siebel application interaction 77

Oracle Business Intelligence Publisher Add-in for Microsoft Word

about using with Siebel Reports 15

advantages of using 16

Excel Analyzer feature 16

installing 38

menu and toolbar, example of 15

Template Builder feature 16

Template Viewer feature 16

using to create report layout templates 91

P

parameter type

Check box, described 111

Date, described 111

DateTime, described 112

Label, described 112

LOV, described 112

Number, described 112

Text, described 112
 TextArea, described 112

parameterized reports

about 108
 about defining parameters in the report
 template 113
 about referencing parameters in the report
 layout template 113
 creating, process of 108
 example of generating 29

purging reports

automatically purging from the Siebel File
 System 64
 from the Oracle BI Publisher Server 64

purging scheduled reports 64

R

report execution modes comparison, diagram of 78

report generation, copying fonts for 62

report layout templates

creating using Oracle BI Publisher Add-in for
 Microsoft Word 91
 defining for selected records 106
 defining for selected records, about 106
 diagram for uploading to a Siebel
 application 95
 registering 93
 sharing 107
 workflow for uploading to a Siebel
 application 95

report notification

disabling 63
 managing 63

report parameters

about defining in the Siebel application 113

report sequence order, changing in the Reports menu 114

report templates

about multiorganizational templates 88
 visibility across organizations, scenario
 for 88

report translation, about 72

reports

about parameterized reports 108
 associating registered reports with application
 views 97
 creating 89
 deleting 24
 determining the report locale at run time 72
 determining whether to modify existing or
 creating custom 91

generating in Siebel Business
 Applications 19

generating sample XML data files for 103
 generating using the Siebel high-interactivity
 client, about 14

generating using the Siebel Open UI
 client 20

generating using the Siebel Open UI client,
 about 13

parameter types and attributes, about 111

registering report layout templates 93

reporting across multiple Siebel business
 objects 99

scenario for creating custom 12

setting user preferences for Siebel Reports at
 the application level 19

troubleshooting 129

uploading report layout templates to a Siebel
 application, workflow for 95

using master-detail reports 115

viewing details for previously generated
 reports 23

Reports Business Service

about 83

Reports menu

changing the report sequence order 114

renaming menu items 114

sample diagram of 14

using to generate reports 14

Run Report pane

using to run reports in Siebel Open UI 13

S

scenario

creating custom reports 12

scheduled reports

adding scrollbars to the Parameters dialog
 box 112

deleting 29

enabling and configuring, process of 57

example of 28

monitoring and viewing 29

suspending and resuming 29

selected records

defining report layout templates for 106

defining report layout templates for,
 about 106

Siebel application-Oracle BI Publisher interaction 77

Siebel Open UI client, using to generate reports 20

Siebel Reports

about 11

- about controlling access to 87
- about integration objects for 82
- about using 11
- administering 61
- architecture, described 77
- development environment, described 77
- directory structure, described 86
- file types, described 83
- overview of 11
- report execution in connected mode 79
- report execution in disconnected mode 81
- report output files types 14
- uploading preconfigured reports to Oracle BI Publisher Server 56
- users, described 11
- using integration object search specifications 99
- using Oracle Business Intelligence Publisher Add-in for Microsoft Word with 15
- workflow for generating reports in connected mode 79
- workflow for generating reports in disconnected mode 81

Siebel Server components, enabling for Siebel Reports 42

T

troubleshooting reports

- alignment problems in JPN and PSJ environments 137
- class not found errors when previewing reports in Microsoft Word 133
- class not found errors when previewing reports in Microsoft Word, guidelines for 132
- column alignment problems in HTML and EXCEL 137
- error after clicking Submit to generate a report 135
- error due to special character violations 136
- error messages for 134
- error while uploading files to Oracle BI Publisher 138
- errors relating to the Siebel File System or

- XMLP directory 136
- errors while running 134
- incorrect path settings in configuration files 135
- Oracle BI Publisher Server login errors 135
- report definition errors 137
- XLIFF errors 135

U

upgrading

- Oracle BI Publisher to latest patch 41

uploading files to Oracle BI Publisher error

- SBL-EAI-04308 138
- SBL-EAI-05010 138

user preferences

- setting at the application level for Siebel Reports 19
- setting report parameters to change the default report output type 19

using Siebel Reports, requirements for 32

W

Web service

- PublicReportServiceService Web service, about 80

WebLogic, starting and stopping 40

workflow

- generating reports in connected mode 79
- generating reports in disconnected mode 81

X

XLIFF (XML Localization Interchange File Format), described 83

XML (Extensible Markup Language), described 83

XPath (XML Path Language), described 83

XSL (Extensible Stylesheet Language), described 83

XSL-FO (Extensible Stylesheet Language-Formatting Objects), described 83

XSLT (Extensible Stylesheet Language Transformation), described 83