Siebel Reports Guide

Siebel Innovation Pack 2015
May 2015

ORACLE®

Contents

Chapter 5:   Administering Siebel Reports

Chapter 6:   Customizing Siebel CRM Reports

Chapter 7:   Customizing Master-Detail Reports

Chapter 8:   Deploying Reports

Chapter 9:   Troubleshooting Siebel Reports

Index

1 What's New in This Release

What's New in Siebel Reports Guide, Siebel Innovation Pack 2015

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 Innovation Pack 2015 is a continuation of the Siebel 8.1/8.2 release.

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 9
- Integrating Oracle BI Publisher with Siebel Business Applications on page 16
- Overview for Using This Book on page 16

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 4, "Running Reports".
- In Microsoft Word to create and customize report templates. For more information, see "About Oracle BI Publisher Desktop" on page 14.
- 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 14.

■ **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 29.

■ **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 6, "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 14.

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 4, "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. The menu that Siebel CRM displays depends on which of the following clients you use:

■ **Siebel Open UI.** Siebel CRM displays the Run Report pane.

■ **High interactivity.** Siebel CRM displays the Reports menu.

The work you do to run a report varies slightly depending on the client you use. The functionality of the Reports menu is nearly identical to the functionality of the Run Report pane. This book describes how to run reports in Siebel Open UI. For brevity, this book uses the phrase Run Report pane, unless noted otherwise. For more information about running a report, see Chapter 4, "Running Reports."

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

Table 2 describes some of the differences with running reports in high interactivity compared to running reports in Siebel Open UI.

Table 2.   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.

Also, you can do the following work in Open UI but not in high interactivity:

■ Use the Run Report pane to override the default settings for the file output type and locale.

■ Click My Reports in the Run Report pane to view reports that have already run. Siebel CRM displays these reports in the My Reports view in the BI Publisher Reports screen.

For more information about using Siebel Reports in Siebel Open UI compared to high interactivity, see 1493786.1 (Article ID), Siebel Reports in Open UI, on My Oracle Support.

For information about using Siebel Open UI, see *Siebel Fundamentals for Siebel Open UI*. For information about using high interactivity, see *Siebel System Administration Guide*.

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 on the left side of a view. It remains open while you navigate through the client.
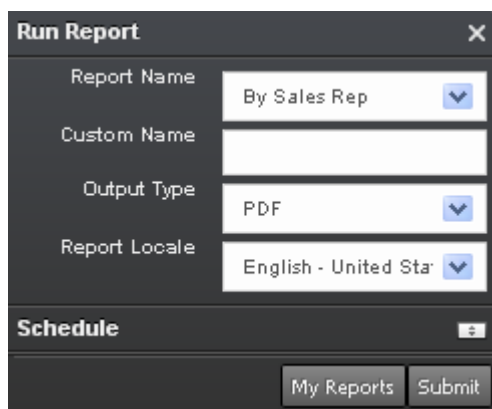
Figure 1.   Run Report Pane in the Siebel Open UI Client

Reports Menu in High Interactivity

Figure 2 includes the Reports menu that high interactivity displays in the client. It displays this menu in a pop-up list. It hides this menu after you choose a menu item from the list.

Figure 2.   Reports Menu in the High-Interactivity Client

How Siebel CRM Displays Reports According to the View

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 19.

The business user determines the report output file types that are available. For more information, see "Registering Report Templates" on page 61.

How Siebel CRM Creates Reports

Oracle BI Publisher is the reporting module that Siebel CRM uses to run and manage reports. Siebel CRM installs OC4J (Oracle Application Server Containers for Java Platform, Enterprise Edition) 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).

■ **OC4J.** See *Oracle Business Intelligence Publisher Installation Guide*, Release 10.1.3 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.
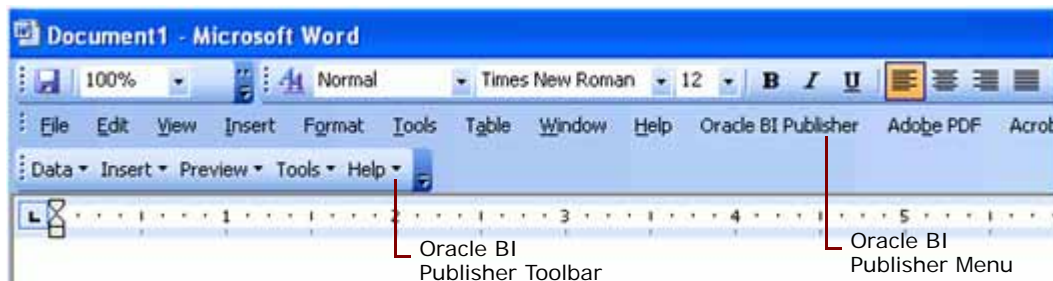
Figure 3.   Oracle BI Publisher Toolbar in Microsoft Word

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 59.

- **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 19.

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).

Integrating Oracle BI Publisher with Siebel Business Applications

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 if 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. For information about how to do this integrating, see 1501378.1 (Article ID) on My Oracle Support.

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 19.

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 and the high-interactivity client. Unless noted otherwise, the information that this book describes applies to Siebel Open UI and high interactivity.

■ 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 SRF (Siebel Repository File) 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.

3 Siebel Reports Architecture

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

- About the Siebel Reports Architecture on page 19
- How Siebel CRM Controls Access to Reports on page 29

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 20
- "How Siebel CRM Runs Reports in Connected Clients" on page 21
- "How Siebel CRM Runs Reports in Disconnected Clients" on page 23
- "How Siebel Reports Uses Integration Objects" on page 24
- "Computer Languages That Siebel Reports Uses" on page 25
- "About the Report Business Service" on page 26
- "Directory Structure That Siebel Reports Uses" on page 28

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

Figure 4.   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:

1 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

2 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.

3 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 13.

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 5 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 5.   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 22.

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 23.

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/SIEBELCRMREPORTS

■ 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 28.

How Siebel CRM Runs Reports in Disconnected Clients

Figure 6 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.
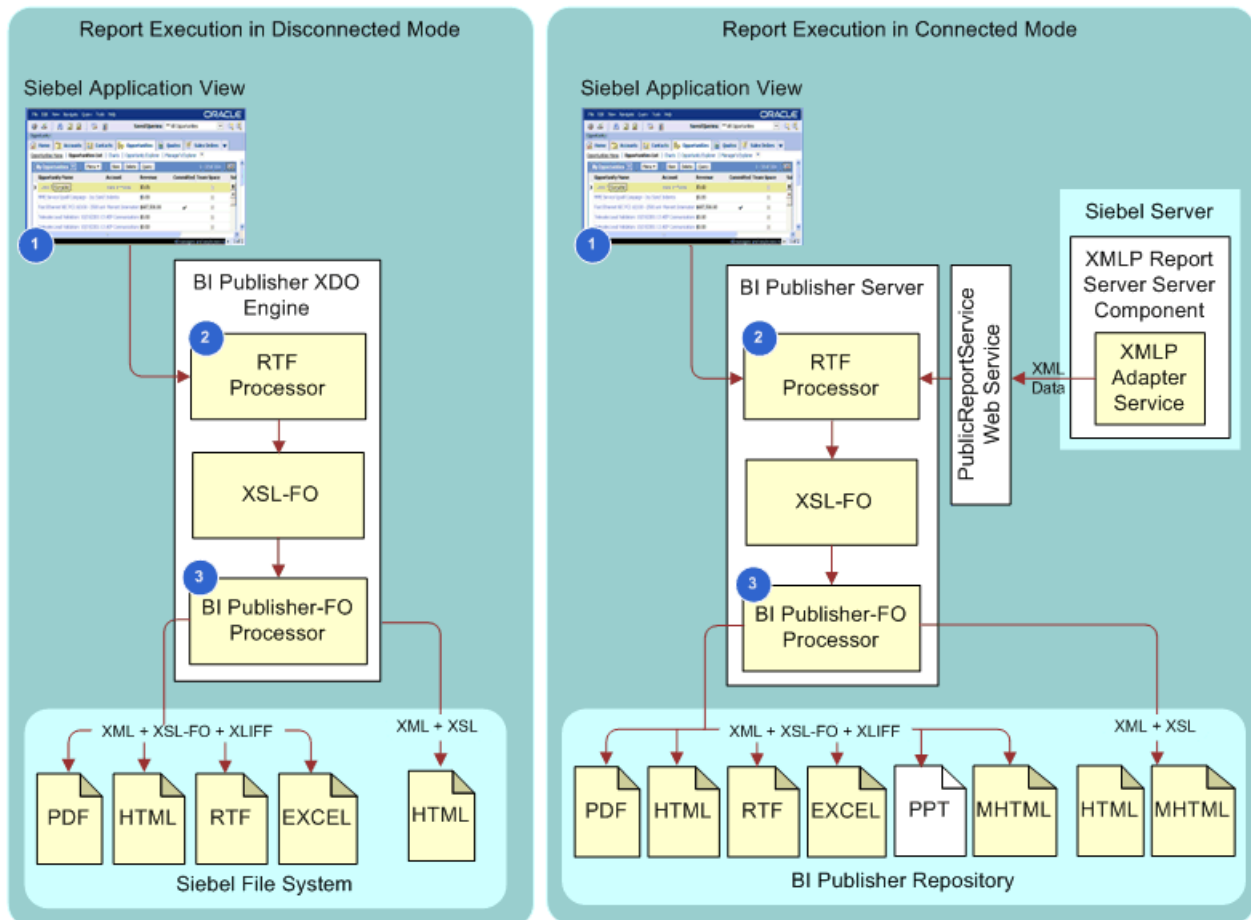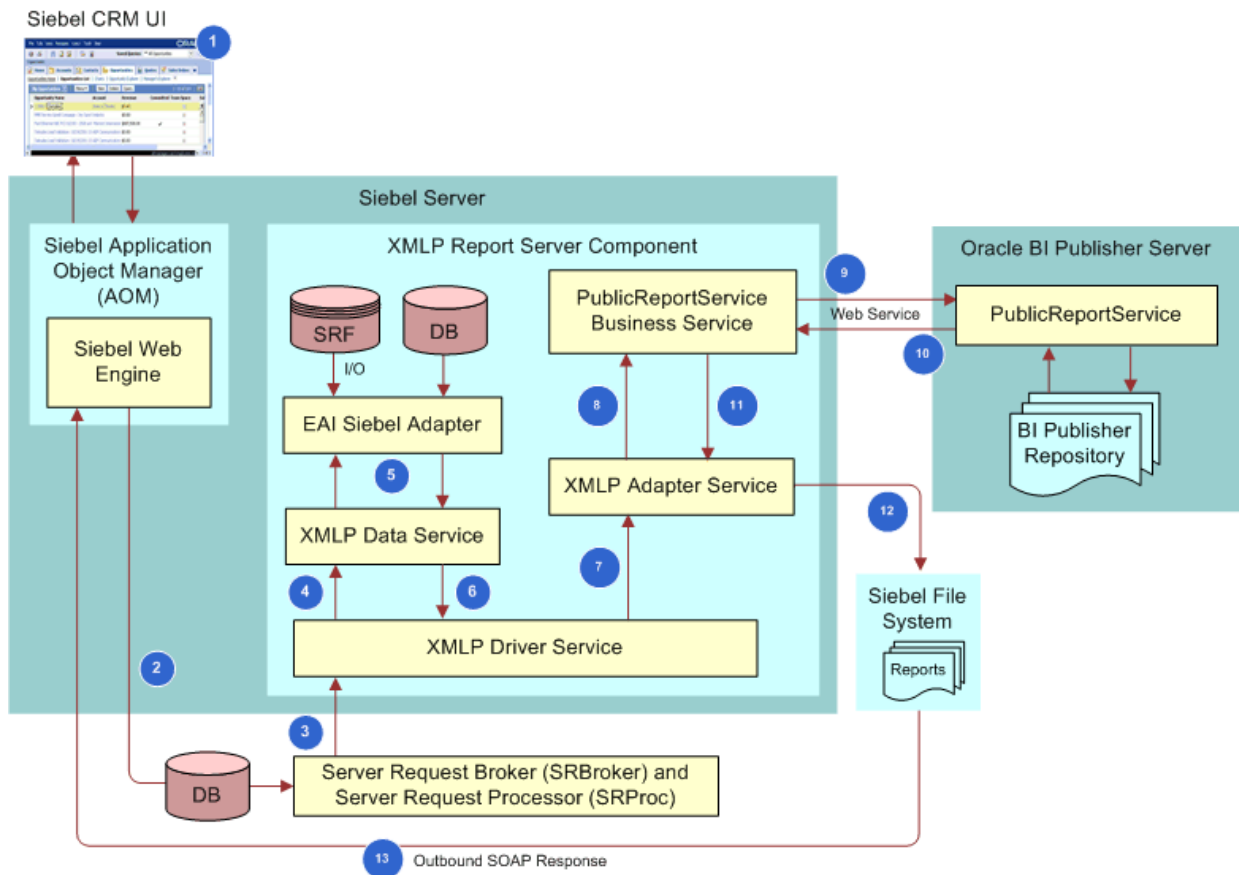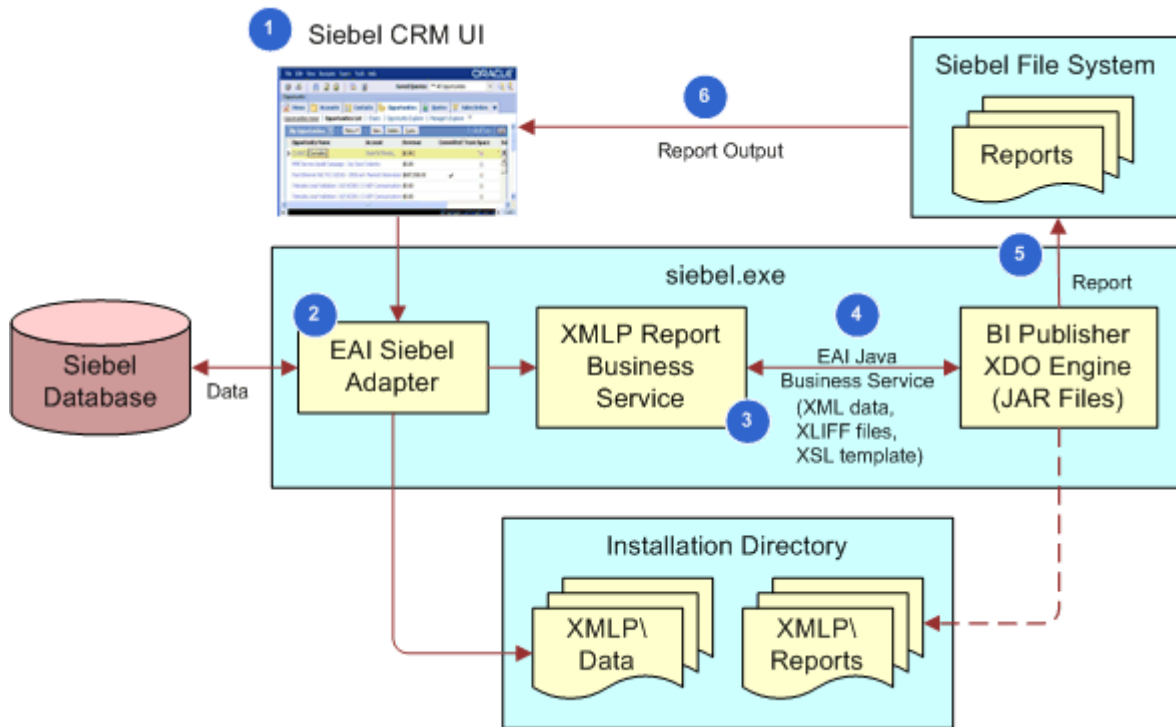

Figure 6.   How Siebel CRM Runs Reports in Disconnected Clients

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 69 and "Creating New Integration Objects to Add Fields to Reports" on page 71.

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 23.

■ **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 13.

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 8 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 3 describes the methods that you can use with the XMLP Driver Service of the Report Business Service.

Table 3.   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.a
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	Sends input arguments to the Reports Business Service that determines how to do one of the following: ■ Store the report in the Report Output business component ■ Attach the report to an Attachment business component. The AttachEntity method is part of the GenerateBIPReport method.

Directory Structure That Siebel Reports Uses

Table 4 describes the folders that Siebel Reports uses. These folders reside in the following folder on the Siebel Server:

SIEBEL_SERVER_ROOT\XMLP

Table 4.   Siebel Reports Directory Structure

Directory	Description
SIEBEL_SERVER_ROOT\XMLP\DATA	Contains the XML files that Siebel CRM creates from integration objects. For more information, see "How Siebel Reports Uses Integration Objects" on page 24.
SIEBEL_SERVER_ROOT\XMLP\FONTS	Contains the predefined .ttf (TrueType format) font files. For more information, see "Making Sure Fonts Are Available for Reports" on page 44.
SIEBEL_SERVER_ROOT\XMLP\REPORTS	Contains the temporary files that Siebel CRM uses to run a report.
SIEBEL_SERVER_ROOT\XMLP\TEMPLATES	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 19.
SIEBEL_SERVER_ROOT\XMLP\xliff*language_code*	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 xliff folder for each language. It uses a language code to identify the language that this language subfolder contains. For example: ■ XLIFF files for German reside in the xliff\deu folder. ■ XLIFF files for French reside in xliff\fra folder. Siebel CRM registers a report template, and then stores the XLIFF file in an *SIEBEL_SERVER_ROOT*\XMLP*language_code* 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.

Directory Structure That the Oracle BI Publisher Repository Uses

Table 5 describes example folders that the Oracle BI Publisher Repository uses with a connected client.

Table 5.   Siebel Reports Directory Structure

Example Directory	Description
ORACLE_HOME/xmlp/XMLP/ Reports/SiebelCRMReports/Account List/	This directory contains the following files: ■ 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: ■ 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 76.

■ 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.

4 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 31
- Scheduling Reports on page 37
- Using Report Parameters to Filter Reports on page 41

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 31
- Running Reports on page 32
- Monitoring Report Status on page 34
- Viewing Report Output on page 35
- Deleting Reports on page 36

The work you do to run a report varies slightly depending on if you use a Siebel Open UI client or a high-interactivity client. This book describes how to run reports in Siebel Open UI. For more information, see "Differences With Running Reports in Siebel Open UI Compared to High Interactivity" on page 11.

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 54.

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 13 and "Types of Users Who Use Siebel Reports" on page 9.

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 31. For information about how Siebel CRM uses multiple languages, see "How Siebel CRM Runs Multilingual Reports" on page 54.

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 34.

■   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 34.

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 9.

High interactivity does not display the Run Report pane. It displays a dialog box. For more information, see "Differences With Running Reports in Siebel Open UI Compared to High Interactivity" 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 12. For information about entering a custom name, see "Entering a Custom Name" on page 34.

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 31 and "How Siebel CRM Runs Multilingual Reports" on page 54.

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 24.

10 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.

If you use the high-interactivity client, then the system administrator must enable report notification. For more information, see "Enabling Report Notification in High Interactivity" on page 44.

11 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 9, "Troubleshooting Siebel Reports."

12 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.

13 (Optional) Use the Print menu to print the report.

14 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 75.

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 75.

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 41.

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 29.

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	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:
	
	 `'Account'.Search = "([Type] LIKE ""Residential*"")"`
	 `'Account'.Sort = "Name(ASCENDING), Location"`
	
	 For more information about search expressions and sort specifications, see *Configuring Siebel Business Applications*.

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 37.

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.

2 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:

■ Starting with Siebel CRM version 8.1.1.10 or version 8.2.2.3 , you can schedule a report in Siebel Open UI, and you can schedule a report that includes parameters. For more information, see "Using Report Parameters to Filter Reports" on page 41.

■ You can schedule a report only in a connected client. For more information, see "How Siebel CRM Runs Reports in Connected Clients" on page 21.

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 6, "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 38.

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 29.

6 (Optional) Enter a custom name.

For information, see "Entering a Custom Name" on page 34.

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 34.

9 View the report output.

For more information, see "Viewing Report Output" on page 35.

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	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: ■ **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. 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). If you click Submit, then Siebel CRM runs the report only one time according to the run date that you specify.

Value	Description
Run Daily/Weekly	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: ■ **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 left Run Time list, and 30 in the right Run Time list, 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. This button is not active if the client uses a light shade of grey for the letters that it displays on this button. ■ **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	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: ■ **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: 2-4 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: 1, 3, 5

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 34.

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 34 and "Viewing Report Output" on page 35.

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 47.

Using Report Parameters to Filter Reports

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 76.

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 34.

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 77.

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.

5 Administering Siebel Reports

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

Doing Typical Administrative Tasks

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

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 26.

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 XLMP 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 that 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 that resides in the Java installation folder for your environment.

Enabling Report Notification in High Interactivity

If your deployment uses the high-interactivity client, then you can enable report notification. Siebel CRM enables report notification in the Siebel Open UI client, by default. You cannot disable report notification in the Siebel Open UI client. 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.

For more information, see "Differences With Running Reports in Siebel Open UI Compared to High Interactivity" on page 11.

To enable report notification in high interactivity

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, and then set the value to one of the following:

■ **On.** Enables report notification.

■ **Off.** Disables report notification.

4 Restart the Siebel Server, and then run a report to make sure notification works correctly.

Changing the Location of the Oracle BI Publisher Repository

Siebel CRM uses the following value to store the location of the Oracle BI Publisher repository in the Oracle home JVM property, by default. If you must change the location of the Oracle BI Publisher Repository for some reason, then you can modify this value:

    ${oracle.home}/xdo/repository

The Siebel application uses this JVM property to access the Oracle BI Publisher repository. If Oracle BI Publisher runs on one of the following items, then the OC4J container automatically sets the oracle.home property value:

■ An Oracle Application server.

■ **OC4J.** For more information about OC4J, see "How Siebel CRM Creates Reports" on page 13.

To change the location of the Oracle BI Publisher repository

■ (Optional) Create a custom XML file:

   a   Copy the xmlp-server-config.xml file to a folder where Oracle BI Publisher can read it.

       For example, in an Oracle AS/OC4J 10.1.3 environment, you copy the configuration file from the following location:

```
OC4J_HOME/j2ee/home/applications/xmlpserver/xmlpserver/WEB-INF/xmlp-server-
config.xml
```

   b   Use a text editor to open the file you created in Step a.

   c   Edit the following default configuration:

```
<?xml version="1.0" encoding="UTF-8"?>
<xmlpConfig xmlns="http://xmlns.oracle.com/oxp/xmlp">
 <resource>
        <file path="${oracle.home}/xdo/repository"/>
 </resource>
</xmlpConfig>
```

   d   To start the application server, add the following JVM option:

```
-Dxdo.server.config.dir=/home/BIP
```

■   (Optional) Replace the oracle.home JVM property name in the xmlp-server-config.xml file with some other JVM property name:

   You must set the value of the new JVM property to the correct repository path.

■   (Optional) Change the absolute path in the WEB-INF/xmlp-server-config.xml file to the repository that you require.

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 36 and "Types of Users Who Use Siebel Reports" on page 9.

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 48

■ "Setting the Polling Interval" on page 49

■ "Setting Concurrency Parameters" on page 49

■ "Optimizing Reports That Include Thousands of Records" on page 50

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 73

■ 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.

High interactivity does not use the waiting period if report notification is enabled. For more information, see "Enabling Report Notification in High Interactivity" on page 44.

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 31.

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. Note that a single Oracle BI Publisher instance that runs on the OC4J can handle approximately 1500 concurrent users For more information about OC4J, see "How Siebel CRM Creates Reports" on page 13.

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 50.

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 (srvrmgr).

To use the Server Manager to set concurrency parameters

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

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

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 37.

Modifying the 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: Operation 'runReport' is expecting a response but no response was
received.
```

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.

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 50 and is applicable with a connected client.

To set server profile parameters

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

For more information about setting this parameter, see 1064043.1 (Article ID) on My Oracle Support. For more information about setting server parameters, see *Siebel System Administration Guide*.

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 20.

To enable scalable mode

1 Log on to the Oracle BI Publisher Server.

2 Navigate to the following folder:

   ORACLE_HOME\xmlp\XMLP\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 through .

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.

3 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 OC4J 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 OC4J process.

To increase memory for the Java Virtual Machine

■ Increase the heap size.

For example, to specify a heap size of 512 megabytes, specify the following value in the oc4j.cmd file:

    –Xms512m –Xmx512m

Different operating systems might use different JVM memory limits.

For more information, see the topic that describes how to set the JVM heap size for OC4J processes in *Oracle® Application Server Performance Guide* in the Oracle Business Intelligence Publisher Documentation Library 10.1.x available on Oracle Technology Network (http://www.oracle.com/technetwork/indexes/documentation/index.html).

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 54

■ "Creating Multilingual Reports" on page 56

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 28.

Priority That Determines the Report Language

Table 6 describes the priority that Siebel CRM uses to determine the report language it uses when it runs a report.

Table 6.   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 4, "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 56.
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 26.

Table 6.   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 31.
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 55. For more information about designing reports, see Chapter 6, "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 55.

To create a multilingual report

1 Design a report in the default language.

ENU (English U.S.) is an example of a default language.

2 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 14.

3 (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 61, 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 28.

  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 65.

■ 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).

6 Customizing Siebel CRM Reports

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

- Process of Creating Custom Reports on page 59
- Specifying the Siebel CRM Data That Report Templates Use on page 66
- Defining Optional Fields When Registering Report Templates on page 73
- Customizing Reports That Use Parameters on page 76
- Customizing Other Options for Siebel Reports on page 82

Process of Creating Custom Reports

To create a custom report, do the following tasks:

1. "Creating Report Templates" on page 59
2. "Registering Report Templates" on page 61
3. "Associating Report Templates with Views" on page 66

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 14.

This task is a step in "Process of Creating Custom Reports" on page 59.

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 24.

2 Install Oracle BI Publisher Desktop.

For detailed information about installing Oracle BI Publisher Desktop and creating a report template, see *Oracle Business Intelligence Publisher Report Designer's Guide* in the Oracle Business Intelligence Publisher Documentation Library 10.1.3.4 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 28.

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

7 Preview the report template:

a In Microsoft Word, open the template file.

Siebel CRM stores the template files in the following folder:

    SIEBSRVR_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 *SIEBSRVR_ROOT*\XMLP\Templates folder.

    ❏ Copy XLIFF (.xlf) files to the *SIEBSRVR_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:

    SIEBSRVR_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 28.

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 59.

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 62.

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 66.

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:

> *SIEBSRVR_ROOT*\XMLP\TEMPLATES

Siebel CRM stores report template files in this folder. For more information, see "Directory Structure That Siebel Reports Uses" on page 28.

  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 61.

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 73.

8 Upload the files:

  ■ If you use a connected client, then click Upload Files.

  ■ If you use a disconnected client, then click Generate Xliff.

For more information about how to upload files, see "How Siebel CRM Uploads Report Templates" on page 64.

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 29.

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 7 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 14.
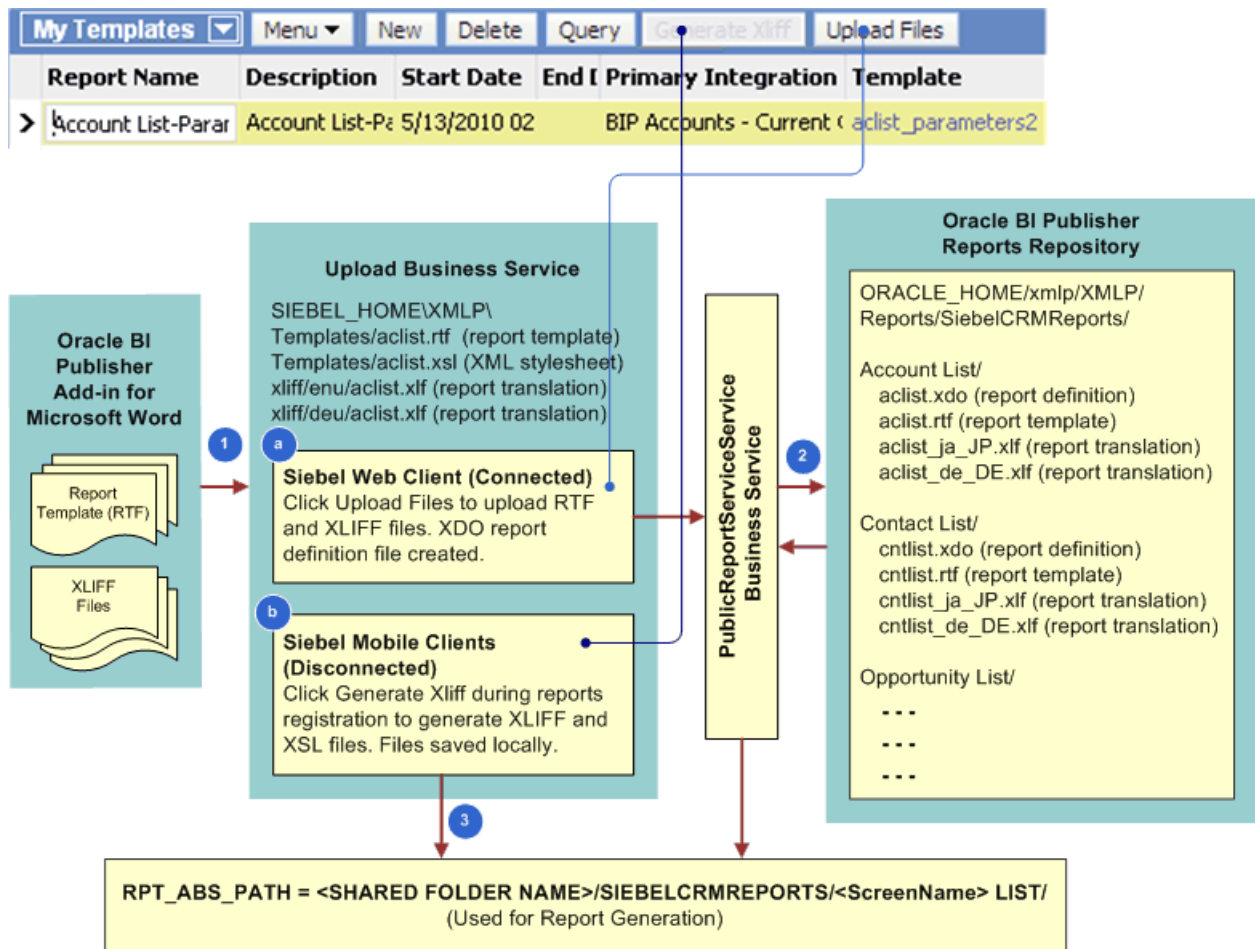

Figure 7. How the Report Template and Translation Files Are Uploaded from Oracle BI Publisher Desktop to Run Reports

Figure 7 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 21.

b In a disconnected client, you click the Generate XLIFF button in the templates list of the Administration - BIP Publisher Reports screen, and then Siebel CRM does the following:

❑ Creates the XSL files.

❑ Does not use the upload service.

If you use a disconnected client to create a report template, then you must copy this report template to the Siebel Server, and you must upload the RTF, XLIFF, and XSL files to the Oracle 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. If you click Upload, then Siebel CRM can use the Oracle BI Publisher Server to run the report in a connected client.

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 23.

Siebel CRM disables the Upload Files button in a disconnected client.

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\xmlp\templates
SIEBEL_ROOT_CLIENT\xmlp\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 12.

This task is a step in "Process of Creating Custom Reports" on page 59.

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 83.

6 Test the new report.

   For more information, see Chapter 4, "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 69

■ "Creating XML Files from Integration Objects" on page 72

■ "Reducing the Amount of Data That Integration Objects Transfer" on page 73

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 24.

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 62.

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 SRF. If no integration object exists that meets your requirements, then see "Modifying Integration Objects" on page 69.
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 67.

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 72.

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 7 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 7.   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.
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 *Configuring Siebel Business Applications*.
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 7:

■ The Position automatically preserves visibility, organization visibility, and view mode. For information, see "How Siebel CRM Controls Access to Reports" on page 29.

■ 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 34.

■ 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 61.

■ 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 72.

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 24.

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 73.

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 62.

 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 Length

Each 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 70 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 you 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 73.

4 Compile your changes.

5 Deploy the integration object:

a In the Integration Objects list, right-click the integration object you located in Step c on page 70, and then click Deploy to Runtime Database.

Siebel CRM overrides the object definition that the SRF file contains for this integration object.

    b Log in to the Siebel client, navigate to the Administration - Web Services screen, and then the Deployed Integration Objects view.

    c Query for the integration object that you deployed in Step a to make sure Siebel CRM deployed it successfully.

    d Navigate to the Administration - Web Services screen, and then Inbound Web Services view.

    e Click Clear Cache to update the run-time database.

6 (Optional) Deploy the modified SRF to the production environment.

For more information, see Chapter 8, "Deploying Reports.".

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 69.

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 72.

■ 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 85.

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 73.

7 Compile your changes.

8 Deploy the integration object:

Do Step 5 on page 70, except deploy the integration object you added in Step 4.

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:

BIPLiteratureFulfillment.xml

5 Save the file in the following folder:

SIEBSRVR_ROOT\XMLP\DATA

You can now create a report template that uses this XML data. For more information, see "Creating Report Templates" on page 59.

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 14. 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 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.

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 75

■ "Sharing Report Templates" on page 76

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 13.

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 55.

       For more information, see "Creating Multilingual Reports" on page 56.

    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:

          SIEBSRVR_ROOT\XMLP\TEMPLATES

       Siebel CRM stores XLIFF files in this folder. For more information, see "Directory Structure That Siebel Reports Uses" on page 28.

       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 75.

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 76.

5 Add report parameters.

For more information, see "Customizing Reports That Use Parameters" on page 76.

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 62.

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 34.

■ **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 29.

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 62.

    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 77

■ "Adding Report Parameters to Report Templates" on page 77

■ "Report Parameter Types You Can Specify" on page 80

■ "Adding Scrollbars to the Parameters Section of the Run Report Pane" on page 81

■ "Guidelines for Defining Parameters in Report Templates" on page 82

About Report Parameters

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 41.

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 37.

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 79.

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 62.

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 above Param2.
	If the Order field is empty, then Siebel CRM displays this report parameter below 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 80.
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 81.

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 80.

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 *Oracle Business Intelligence Publisher Report Designer's Guide* Release 10.1.3.4 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 34.

Report Parameter Types You Can Specify

Table 8 describes the report parameter types that you can specify in the Type field of the Parameters list.

Table 8.   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.
LOV	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.
	If the picklist is:
	■ **Bounded.** The LOV parameter is bounded.
	■ **Not bounded.** The LOV parameter is not bounded.

Table 8.   Report Parameter Types

Parameter Type	Description
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	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. You can use the Label parameter type only with the Order field. Siebel CRM aligns a Label parameter type along the left edge 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. You cannot use the Label parameter type to specify font face, size, color, style, alignment, or wrap in HTML.
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.

7 Compile the SRF.

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:

■ "Renaming Items in the Reports Menu in High Interactivity" on page 82

■ "Modifying the Sequence of Report Names in the Run Report Pane" on page 83

Renaming Items in the Reports Menu in High Interactivity

You can modify the text strings that a high-interactivity client displays in the Reports Menu for the following menu items:

■ My BI Publisher Reports.

■ **Schedule Report.** Siebel CRM displays the Schedule section only if Siebel CRM associates the user with the XMLP_SCHEDULER responsibility, by default. For more information, see "How Siebel CRM Controls Access to Reports" on page 29.

For example, you can modify *My BI Publisher Reports* to *My Reports,* or some other value.

The Open UI client does not use these menu items. For more information, see "Differences With Running Reports in Siebel Open UI Compared to High Interactivity" on page 11.

To rename items in the Reports Menu in high interactivity

1 Log in to Siebel Tools.

2 In the Object Explorer, click Symbolic String.

3 (Optional) Modify an item in the Run Report pane:

    a In the Symbolic Strings list, query for SBL_MY_BIPUBLISHER_REPORTS.

    b Edit the value in the Current String Value property.

4 (Optional) Modify the Schedule section:

    a In the Symbolic Strings list, query for SBL_SCHEDULE_BIPUBLISHER_REPORTS.

    b Edit the value in the Current String Value property.

5 Compile the Symbolic String project.

6 Distribute the updated SRF, and then restart the Siebel Server.

7 Verify your modifications:

    a Log in to the Siebel application, and then navigate to any screen.

    b Click the Reports button in the application toolbar, and then verify your modifications.

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 12.

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.

7 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 8 includes the Service Request Activity - All report in Siebel Service, which is an example of a master-detail report.

Figure 8.   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 86.

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 90.

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 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 24.

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 8 on page 85.

2 Open Oracle BI Publisher Desktop.

For more information, see "About Oracle BI Publisher Desktop" on page 14.

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 28

The following screen capture includes some of the features that this template uses:

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:

(<?if: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 62.

4 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 88.

Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Table 9 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 *Oracle Business Intelligence
Publisher Report Designer's Guide*, Release 10.1.3.4 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 24.

Table 9. Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Syntax	Oracle BI Publisher Code	Description
If	`<?if://ssServiceRequest?>`	An IF condition that determines if service request record exist, do not exist, or are not displayed.
for-eachgroup: ssServiceRequest	`<?for-each-group:ssServiceRequest; position()?>`	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	`<?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.
ssSeverity ssDescription	`<?ssSeverity?>` `<?ssDescription?>`	Field mappings.
If	`<?if://ssClosedDate?>`	If condition that does not display the ssClosedDate field if null.
ssClosedDate	`<?format-date:psfn:totext(ssClosedDate, "yyyy-MM-dd'T'HH:mm:ss", "MM/dd/yyyy hh:mm:ss"); 'SHORT_TIME'?>`	Field formatting.

Table 9.   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 sssStatussType 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 9.   Code That Oracle BI Publisher Uses for the Service Request Activity (All) Report Template

Syntax	Oracle BI Publisher Code	Description
If	`<?if:not(// ssServiceRequest)?>`	If condition that prints "No Data Found" at the bottom of the report if there are no records.
End	`<?end if?>`	End if statement for above condition.

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 24.

■ 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 86.

■ 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 14.

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 86.

9 Register the report.

For more information, see "Registering Report Templates" on page 61.

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.

8 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 93

■ Deploying Integration Objects to the Production Environment on page 98

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 93

2 "Packaging the Report Files in the Development Environment" on page 94

3 "Deploying the Report Files to the Production Environment" on page 95

4 "Packaging Report Database Records in the Development Environment" on page 96

5 "Deploying Report Database Records to the Production Environment" on page 97

6 (Conditional) "Deploying Integration Objects to the Production Environment" on page 98

Preparing the Deployment Environment

This task is a step in "Process of Deploying Reports" on page 93.

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 .

To package the report files in the development environment

1 Manually copy the files that you must deploy from the *SIEBSRVR_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

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 init 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:

    deploy_enterprise load 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

4 Run the following command to create a session for the deployment:

    deploy_enterprise create username password PackageName

5 Run the following copy command to deploy the package:

    deploy_enterprise copy username password PackageName

6 Navigate to the following folder on the Siebel Server:

    siebsrvr\XMLP

7 Verify that the files you placed in the deployment package described in 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 93.

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 96, 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 69.

To deploy integration objects to the production environment

1 Create, and then export an SIF file for the integration object:

a Log in to Siebel Tools.

b In the Object Explorer, click Integration Object.

c In the Integration Object list, right-click the integration object that Siebel CRM must use to create the SIF file, and then click Add to Hot Fix.

d In the Generate Hot-Fix dialog box, enter a label, and then click Export.

Siebel Tools adds a SIF file in the following folder:

  Tools\ADM

2 Copy the SIF and descriptor files in the repository folder of the ADM package that you created when you ran the admpkgr init command.

For more information about running this command, see "Packaging the Report Files in the Development Environment" on page 94.

3 Do the work described in "Packaging the Report Files in the Development Environment" on page 94.

If no report exists for this integration object, and if you do not plan to move a report for this integration object, then you can skip Step 1 on page 94.

4 Verify that Siebel CRM moved the integration object to the production environment:

a Log in to the Siebel client with administrator privileges.

Use a computer that resides in the production environment.

b Navigate to the Administration - BIP Publisher Reports screen, and then the Sample Data File Generation view.

c Make sure Siebel CRM displays the integration object in the Sample Data File Generation list.

For more information about this list, see "Creating XML Files from Integration Objects" on page 72.

9 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 101
- Fixing Class Not Found Errors on page 104
- Fixing Class Not Found Errors When Previewing Reports in Microsoft Word on page 105
- Troubleshooting Error Messages for Siebel Reports on page 107

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 101
- "Enabling Logging and Debugging on the Oracle BI Publisher Server" on page 103
- "Enabling and Disabling Debugging in Disconnected Clients" on page 104

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.

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 Name Server runs.

   ■   *port number* identifies the port number where the Gateway Name Server 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 chapter about setting the system maintenance options in *Oracle Business Intelligence Publisher Administrator's and Developer's Guide*, Release 10.1.3.4 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.

To enable debugging on the Oracle BI Publisher Server

1 Log in to Oracle BI Publisher Server with administrator privileges.

2 Click the Admin tab, and then Server Configuration under System Maintenance.

3 Change the Debug Level from Exception to Debug, and then click Apply.

4 Use a text editor to create a new file named xdodebug.cfg. Save this file in the following folder:

```
ORACLE_HOME\oc4j_b\jdk\jre\lib
```

where:

■ O*RACLE_HOME\oc4j_b* is the folder where you installed Oracle BI Publisher Server. For more information about OC4J, see "How Siebel CRM Creates Reports" on page 13.

5 Add the following code to the file you created in Step 4:

```
LogLevel=log_level
LogDir=logging_folder_path
```

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 jre\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 jre\lib installation folder to another computer.

2 Remove or rename the xdo.cfg that resides in the jre\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\client\classes…
```

■ 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 :

```
echo %1

set _JAVA_OPTIONS=-Xbootclasspath/a:
```

```
C:\81DQSSIA\client\classes\SiebelXMLP.jar;C:\81DQSSIA\client\
classes\siebel.jar;C:\81DQSSIA\client\classes\XSLFunctions.jar;C:\81DQSSIA\
client\classes\SiebelCustomXMLP.jar;C:\81DQSSIA\client\classes\SiebelCustomXMLP
_SIA.jar
```

```
"C:\Program Files\microsoft office\Office\Winword.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 10.

Table 10.   Problems with Siebel Reports

Symptom or Error Message	Solution
SBL-OMS-00203: Error <?> invoking method "<?>" for Business Service "<?>"	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. To fix this problem, see *"Fixing Errors That Occur When Running Siebel Reports (SBL-OMS-00203)" on page 112*.
Siebel CRM displays the following message if you click Submit in the Run Report pane" "Unable to find definition for component XMLPReportServer(SBL-SRQ-00103)"	This error might occur after you click Submit to run a report. It might occur in a connected or disconnected client. To fix this problem, see *"Fixing Errors That Occur After Clicking Submit in the Run Report Pane (SBL-SRQ-00103)" on page 112*.
The specialized method 'GetFileToDir' is not supported on Business Component 'Report Template BC' used by Business Object 'Report Administration'. (SBL-DAT-00322)	This error might occur if the path settings in the configuration files are not correct. It might occur in a connected or disconnected client. To fix this problem, see *"Fixing Errors That Occur Due to Incorrect Configuration File Settings (SBL-DAT-00322)" on page 112*.
SBL-RPT-50504: Error occurred while saving the record. No data file provided for XLIFF generation.	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. To fix this problem, see *"Fixing XLIFF File Errors" on page 114*.
Siebel CRM displays the following message if you run a report: Class name incorrect	This error might occur if Oracle BI Publisher or the JAR files are not loaded. It occurs only in a disconnected client. To fix this problem, you set the classpath. For more information, see *"Fixing the CLASSPATH Settings" on page 114*.

Table 10. Problems with Siebel Reports

Symptom or Error Message	Solution

Siebel CRM does not create reports that include a custom java class.	Verify that the SiebelCustomXMLP.jar and SiebelXMLP.jar files reside in the following folder:
	
	  SIEBSRVR_ROOT\classes
	
	Make sure that the Oracle BI Publisher Server path is the following:
	
	  $OC4J_PATH\j2ee\home\applications\xmlpserver\xmlpserver\WEB-INF\lib
	
	For Siebel Industry Applications, make sure the following files reside in each of these paths:
	
	■ SiebelCustomXMLP_SIA.jar
	
	■ SiebelXMLP.jar
	
	This problem occurs only in a connected client.

The Oracle BI Publisher log file or console displays the following message:
 	
  ][java.lang.String][EXCEPTION]
  oracle.apps.xdo.XDOException: XSLT10gR1: Failed Secure Java Extensions check.
 	
This error occurs if you run a report and:
 	
■ You do not copy the JAR files to the Oracle BI Publisher Server.
 	
■ The Disable External Reference parameter attribute is not set to FALSE.
	To fix this problem, see the topic on starting and stopping OC4J and Oracle BI Publisher in 1501378.1 (Article ID) on My Oracle Support.

SBL-EAI-04116 BIP: Server connection is lost	This error might occur with a very large or complex report. It occurs only in a connected client. You can increase the HTTPSleepTime method argument to fix this problem. For more information, see "Modifying the Sleep Time" on page 51.

Table 10.   Problems with Siebel Reports

Symptom or Error Message	Solution
Siebel CRM displays one of the following messages in the XMLP log file: ■ `Verify the BIP Server Userid and Password. (SBL-RPT-50529)` ■ `Failed to log into BI Publisher: invalid username or password.`	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. To fix this problem, see "Fixing Oracle BI Publisher Server Login Errors" on page 114.
Generic Error in Compression Routine (SBL-UIF-00227)	Make sure the `temp\xmlp` folder exists in the *SIEBSRVR_ROOT* folder. If it does not, then add it. For more information, see "Directory Structure That Siebel Reports Uses" on page 28. Also, make sure the Siebel File System is set correctly. For information about setting the Siebel File System, see *Siebel System Administration Guide*. This problem might occur in a connected or disconnected client.
PublicReportService::executeCreate Report Failure: due to Report Name contains special characters for report	Make sure the report name does not contain the following special characters: ■ Forward slash (/) ■ Backslash (\) ■ Double quote (") ■ Single quote (') ■ Tilde (~) ■ Ampersand (&) ■ Asterisk (*) ■ Plus sign (+) ■ Left angle bracket (<) ■ Right angle bracket (>) ■ Percent sign (%) This problem occurs only in a connected client.

Table 10.   Problems with Siebel Reports

Symptom or Error Message	Solution
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".	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. 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 61. This problem occurs only in a connected client.
■ An column alignment problem occurs in HTML report output.	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. This problem might occur in a connected or disconnected client. To fix this problem, see "Fixing Alignment Problems in HTML" on page 114.
One of the following problems occur: ■ An alignment problem occurs in report that runs in Japanese or PSJ. ■ The PPT output type for Japanese (JPN) is distorted in a report.	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. To fix this problem, see "Fixing Japanese Font Errors" on page 115.

Table 10.   Problems with Siebel Reports

Symptom or Error Message	Solution

Siebel CRM displays one of the following message when you upload files to Oracle BI Publisher:
 	
■ SBL-EAI-05010: Class name incorrect or does not extend SiebelBusinessService : com/siebel/data/SiebelPropertySet -- JVM Exception:java.lang.NoClassDefFoundError: com/siebel/data/SiebelPropertySetObjMgrBusServiceLog
 	
■ Object manager error: ([0] Class name incorrect or does not extend SiebelBusinessService : <?>
	To fix this problem, verify the following:
	
	■ JAVA_HOME variable is set correctly.
	
	■ The following JAR files are copied from the *siebel root*\classes\original folder to the *ORACLE_HOME*\j2ee\home\applications\xmlpserver\xmlpserve\WEB-INF\lib folder:
	
	   ■ 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.

Siebel CRM displays the following message when you upload files to Oracle BI Publisher:
 	
   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::executeUploadReport Failure: Due to Report with Path [/SiebelCRMReports/Application Activity/Application Activity.xdo] already exist!".
	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.
	
	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:
	
	   ORACLE_HOME\xmlp\XMLP\Reports\SiebelCRMReports

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 jre\lib folder.

For information about doing these tasks, see "Enabling Logging for Siebel Reports in the Siebel Application" on page 101 and "Enabling and Disabling Debugging in Disconnected Clients" on page 104.

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 61.

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 is shared.

3 In the XMLPReports section of the .cfg file, make sure the following section is defined:

```
[XMLPReports]
XdoDir = /xmlp/templates/
ReportOutputDir = /xmlp/reports/
ReportDataDir = /xmlp/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 .

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 srvrmgr program, see *Siebel System Administration Guide*.

Run the following command to change the CLASSPATH:

```
change param CLASSPATH=SIEBSRVR_ROOT\classes\Siebel.jar;
SIEBSRVR_ROOT\classes\SiebelXMLP.jar; SIEBSRVR_ROOT\classes\wlfullclient.jar;
```

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 folder on the computer where you installed the Oracle BI Publisher Server:

　　\Java\jdk\jre\lib\fonts

3 Restart OC4J.

For more information about OC4J, see "How Siebel CRM Creates Reports" on page 13.

4 Log in to Oracle's BI Publisher Server.

5 Navigate to Admin, Runtime configuration, and then Font Mappings.

6 Create the following font mappings:

■ Arial, Normal, msgothic.ttc

■ Arial, Bold, msgowthic.ttc

Index