

Siebel

Reports Guide

June 2026



Siebel
Reports Guide

June 2026

F87470-07

Copyright © 1994, 2026, Oracle and/or its affiliates.

Author: Sukanya Mishra, Carmen Myrick

Contents

Get Help i

1 What's New in This Release 1

- What's New In Siebel Reports Guide, Siebel CRM 21.5 Update 1
- What's New In Siebel Reports Guide, Siebel CRM 20.1 Update 1

2 Overview of Siebel Reports 3

- Overview of Siebel Reports 3
- Conventions Used in This Guide 3
- Output File Types That Siebel Reports Supports 4
- Types of Users Who Use Siebel Reports 5
- How Siebel Creates Reports 5
- Where to Find More Information About Oracle Analytics Publisher 5
- Oracle Publisher Desktop 6

3 Running Reports 9

- Running Reports 9
- Running Reports in Siebel Clients 9
- Scheduling Reports 16

4 Integrating Oracle Publisher with Siebel 21

- Integrating Oracle Publisher with Siebel 21
- About Integrating Oracle Publisher with Siebel 21
- Requirements for Using Siebel Reports for First-Time Installations 22
- Roadmap for Integrating Oracle Publisher with Siebel for First-Time Installations 23
- Preparing to Integrate Oracle Publisher with Siebel 23
- Process of Installing Oracle Publisher for Integration with Siebel 24
- Upgrading to the Latest Oracle Publisher Patch 28
- Configuring an Outbound Web Service for Siebel Reports 28
- Enabling Siebel Server Component Groups for Siebel Reports 29

Configuring Security and Authentication for Siebel Reports	30
Configuring Security and Authentication for Siebel Reports Using LDAP	34
Configuring Siebel Reports in a Web Single Sign-On Environment	37
Adding Explicit Reference to Container URL for Oracle Publisher Server	42
Implementing EAI File Handling Changes for Siebel Reports Generation	42
Uploading Preconfigured Report Layout Templates to Oracle Publisher Server	43
Process of Enabling and Configuring Report Scheduling	44
About Upgrading Siebel Reports for Siebel 17.0	46

5 Administering Siebel Reports 47

Administering Siebel Reports	47
Doing Typical Administrative Tasks	47
Optimizing the Performance of Siebel Reports	50
Optimizing Reports That Include Thousands of Records	52
Administering Multilingual Reports	56
Creating Multilingual Reports	57
Administering Signatures in Reports	60

6 Siebel Reports Architecture 63

Siebel Reports Architecture	63
About Siebel Reports Architecture	63
About How Siebel Runs Reports in Clients	63
How Siebel Reports Uses Integration Objects	68
Computer Languages That Siebel Reports Uses	68
Directory Structure That Siebel Reports Uses	69
Directory Structure That Oracle Publisher Repository Uses	70
How Siebel Controls Access to Reports	70
How Siebel Uses Report Templates With Organizations	71

7 Siebel Reports Business Service 73

Siebel Reports Business Service	73
Guidelines for Modifying the Reports Business Service	73
Activate the Siebel BI Publisher 8 Reports Business Service: Quick Steps	74
Test the Siebel BI Publisher 8 Reports Business Service Setup	75
Activate the Siebel BI Publisher 8 Reports Business Service: Overview of the Complete Steps	75
Activate Siebel BI Publisher Reports Business Service (8.1.1.7+)	92

8	Customizing Siebel CRM Reports	95
	Customizing Siebel Reports	95
	Process of Creating Custom Reports	95
	Specifying the Siebel Data That Report Templates Use	100
	Defining Optional Fields When Registering Report Templates	107
	Customizing Reports That Use Parameters	109
9	Customizing Master-Detail Reports	117
	Customizing Master-Detail Reports	117
	About Master-Detail Reports	117
	Viewing the Report Template of a Master-Detail Report	118
	Creating a Master-Detail Report Template in Microsoft Word	121
10	Deploying Reports	123
	Deploying Reports	123
	Process of Deploying Reports	123
	Deploying Integration Objects to the Production Environment	127
11	Troubleshooting Siebel Reports	129
	Troubleshooting Siebel Reports	129
	Enabling and Disabling Logging for Siebel Reports	129
	Fixing Class Not Found Errors	132
	Troubleshooting Error Messages for Siebel Reports	133
12	Appendix: List of Siebel Reports	141
	Siebel Web Client Reports	141
	Siebel Mobile/Sample Database Reports	146

Get Help

Preface

This preface introduces information sources that can help you use the application and this guide.

Using Oracle Applications

To find guides for Oracle Applications, go to the Oracle Help Center at <https://docs.oracle.com/>.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the [Oracle Accessibility Program website](#).

Contacting Oracle

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit [My Oracle Support](#) or visit [Accessible Oracle Support](#) if you are hearing impaired.

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send an email to:
oracle_fusion_applications_help_ww_grp@oracle.com.

1 What's New in This Release

What's New In Siebel Reports Guide, Siebel CRM 21.5 Update

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

As of Siebel CRM 21.5 Update, Oracle uses Oracle® Business Intelligence Publisher (Oracle BIP) or Oracle® Analytics Publisher (OAP) as the Reporting Engine. In this guide, the use of the following terms refers to both Oracle BIP and OAP: *Publisher, Publisher Desktop, Publisher Server, Publisher Enterprise, Reports Publisher*, and so on.

What's New In Siebel Reports Guide, Siebel CRM 20.1 Update

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

2 Overview of Siebel Reports

Overview of Siebel Reports

This chapter provides an overview of Siebel Reports.

It includes these topics:

- *Conventions Used in This Guide*
- *Output File Types That Siebel Reports Supports*

Conventions Used in This Guide

Familiarize yourself with some conventions used in this guide.

Terms

This book uses the following terms, unless noted otherwise:

- The *client* is the Siebel CRM client that a Siebel Business Application uses. This book describes how to use the Siebel Open UI client. Unless noted otherwise, the information that this book describes applies to Siebel Open UI.
- A *user* is a person who uses the client.
- The *server* is the Siebel Server.
- An *administrator* is anyone who uses an administrative screen in the client to configure Siebel Reports. The Administration - Server Configuration screen is an example of an administrative screen.
- A *predefined object* is an object that comes already defined with Siebel CRM. The objects that Siebel Tools displays in the Object List Editor immediately after you install Siebel Tools and the Siebel runtime repository but before you make any customization are predefined objects.
- The term *focus* indicates the currently active object in the client. To indicate the object that's in focus, Siebel CRM typically sets the border of this object to a solid blue line.

Fonts

Computer font: This font typically indicates a value you enter or text that Siebel CRM displays. It's also used for code snippets and file or server paths.

`This is computer font.`

Code block font: Code block font is used for longer code examples. It looks like this:

`Code block font looks like this. In HTML, it includes a convenient copy button you can use the copy the code example.`

Business Objects

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 in this guide assume that you don't use the Tree navigation control option to access screens and views. However, you can use the Tree navigation control if you need to. For more information about navigation controls, see the *Siebel Fundamentals Guide*.

Output File Types That Siebel Reports Supports

Users can save a report as one of these file types:

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

Note: An XLF file is required to upload the Excel template. The XLF file is used by the publisher to translate English to non-ENU languages. Customers who want to use a non-ENU language in their Excel reports must use RTF to create an XLIFF, because only RTF and XPT generate XLIFF. This is a workaround, since, in Excel layouts, XLIFF isn't supported. The XLIFF must be attached along with the Excel Report template in the BIP Reports Admin Screen and uploaded to BIP\OAP.

- PPT (Microsoft PowerPoint format)
- MHTML (MIME HTML format)
- PPTX (Microsoft PowerPoint format, supported as of Siebel CRM 21.5 Update)
- OAP supports many more output formats. For more information, see Output Types in <https://docs.oracle.com/en/middleware/bi/analytics-server/user-publisher-oas/using-oracle-analytics-publisher-oracle-analytics-server.pdf>.

The PPT and MHTML report output file types are available only with a connected client. For more information, see *About Siebel Reports Architecture*.

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

Types of Users Who Use Siebel Reports

The following types of users can use Siebel Reports:

- **Reports user.** Runs, views, monitors, schedules, and deletes reports. For example, the report user in a sales organization might include a salesperson, sales manager, or sales executive. If this user uses a computer that includes the Oracle Analytics Publisher Add-in installed in Microsoft Word, then this user can modify an existing or create a custom report template. For more information, see [Oracle Publisher Desktop](#).
- **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 Controls Access to Reports](#).
- **Report administrator.** Installs and configures Siebel Reports, deletes reports, and manages how Siebel CRM translates reports from one language to another language.

How Siebel Creates Reports

Oracle Publisher (BIP or OAP) is the reporting module that Siebel CRM uses to run and manage reports. Siebel integrates with Oracle Publisher for report execution and administration.

Oracle WebLogic Server and Oracle Publisher Server are installed as part of Oracle Publisher installation.

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

For more information about Oracle Publisher, see the documentation available at [Oracle Help Center/Middleware/BI](#).

Where to Find More Information About Oracle Analytics Publisher

To find more information about Oracle Analytics Publisher, see:

- Downloading Oracle BI Publisher/Oracle Analytics Publisher: See [Oracle Analytics Publisher Downloads](#).
- General information about Oracle Analytics Publisher: See the [Oracle Analytics Publisher](#) site.

Oracle Publisher Desktop

Oracle Publisher Desktop is a Microsoft Word add-in that you can use to create a report template, which is saved as an RTF file. Oracle Publisher Desktop is also known as the Oracle Publisher Add-in for Microsoft Word.

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.

The publisher add-in includes layout, query, and language functionality. It separates logic from the layout, and the layout doesn't depend on a specific language. The add-in provides flexibility during deployment and helps to reduce maintenance costs. It also includes documentation, demonstrations, and examples.

Microsoft Word displays the Publisher toolbar and Publisher menu after you install Oracle Publisher Desktop.

Note the following:

- As a best practice, only use this publisher to create report templates. If you use some other tool to create report templates, then you must manually code the XSL statements and references to the data fields. This book doesn't describe how to code XSL statements.
- Use the same version of Oracle Publisher Server and Oracle Publisher Desktop in your environment.
- Siebel CRM lets you use only some functionality that Oracle Publisher Desktop provides. Oracle only supports the features that this guide describes.

For more information see [Oracle Help Center/Middleware/BI](#).

Oracle Publisher Desktop Features

Oracle Publisher Desktop provides the following features:

- **Template Builder.** Lets you modify and customize report templates. Users 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's 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 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](#).

- **Template Viewer.** Enables you to use sample data to preview a report in multiple output formats.
- **Excel Analyzer.** Lets you do the following work:
 - Export report query results to an Excel spreadsheet.
 - Sign in to Oracle 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 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 Siebel Reports Architecture](#). Excel Analyzer is deprecated in Oracle Analytics Publisher, but for instructions on how to reuse old analyzer templates,

see *Oracle Analytics Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server* available at <https://www.oracle.com/middleware/technologies/analytics-publisher/downloads.html>.

For more information about using Oracle Publisher Desktop, see *Report Designer's Guide for Oracle Business Intelligence Publisher* and *Oracle Analytics Publisher documentation* available at <https://docs.oracle.com/en/middleware/bi/index.html>.

Advantages of Using Oracle Publisher Desktop

Oracle 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 novice users to create a report template in Microsoft Word, and then lets a business user use this template to finalize a 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.
- Lets users get data, and to get it from multiple sources.
- Allows a business user and a 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.

3 Running Reports

Running Reports

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

- *Running Reports in Siebel Clients*
- *Scheduling Reports*
- *Using Report Parameters to Filter Reports*

Running Reports in Siebel Clients

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

- *Setting User Preferences for Siebel Reports*
- *Running Reports*
- *Monitoring Report Status*
- *Viewing Report Output*
- *Deleting Reports*

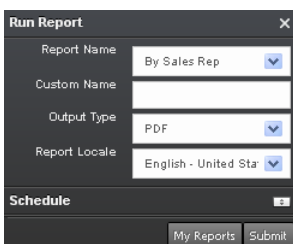
How Users Run Reports

To run reports, users typically:

1. Go to a view.
2. Run a search.
3. Select the Report button in the application toolbar.
4. Select a report to run.

Users can run the report immediately or schedule it to run at a later time. They can monitor reports while they run, view prior reports, or delete prior reports. For more information about running reports, see *Running Reports*.

You use the Run Report pane, shown in the following image, to run reports in Siebel Open UI. The Report Name field in the Run Report pane displays the reports that you can run for a view. The Run Report pane remains open while you navigate through the client.



Scenario for Customizing Reports

Different types of users might customize reports in different ways, depending on the type of user and the user's needs.

For details about customizing reports, see *Customizing Siebel 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 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 *Oracle Publisher Desktop*.

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 *Running Reports*.

Setting User Preferences for Siebel Reports

A user preference that you set overrides the configuration that exists in the report templates.

1. Sign in to the Siebel client.
2. Navigate to the User Preferences screen, and then to 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 Runs Multilingual Reports*.

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](#) and [Types of Users Who Use Siebel Reports](#).

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](#). For information about how Siebel CRM uses multiple languages, see [How Siebel Runs Multilingual Reports](#).

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](#).

- 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](#).

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](#).

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 Displays Reports According to the View](#). For information about entering a custom name, see [Entering a Custom Name](#).

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* and *How Siebel Runs Multilingual Reports*.

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*.

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

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

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

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

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

12. Click Submit.

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

For more information, see *Managing Report Notifications*.

13. Click My Reports.

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

14. Click Account List in the Report Name field.

Siebel CRM displays the report in the format that you specify in *Running Reports*. In this example, it displays the report in a PDF file. It stores the report in the Siebel file system.

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

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

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

Using the CTRL Key to Choose Multiple Records for a Report

A business user must specify that you can use the CTRL key to choose multiple records for a report. For more information, see *Allowing Users to Choose The Records That a Report Contains*.

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*.

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.

Using Report Parameters to Filter Reports

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

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

Siebel Open UI supports report parameters starting with Siebel CRM versions 8.1.1.10 and 8.2.2.3. For more information, see *Customizing Reports That Use Parameters*.

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*.
 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*.

7. Click Submit.

The Oracle 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.

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*.

To monitor report status

1. Log in to the client.
2. Navigate to the BI Publisher Reports screen, 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 Controls Access to Reports*.

To view report output

1. Log in to the client.
2. Navigate to the BI Publisher Reports screen, 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: <code>'Account'.Search = "([Type] LIKE "Residential*")"</code> <code>'Account'.Sort = "Name(ASCENDING), Location"</code> For more information about search expressions and sort specifications, see <i>Configuring Siebel Business Applications</i> .
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.

Field	Description
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.

How Siebel Displays Reports According to the View

The reports that Siebel CRM displays in the Report Name list in the Run Report pane depends on the which view the client's showing.

For example, if the client is showing:

- 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.

Users can specify the reports that Siebel CRM displays.

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](#).

To delete a report

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

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

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

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

Scheduling Reports

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

- You can schedule a report in Siebel Open UI. You can also schedule a report that includes parameters. For more information, see [Using Report Parameters to Filter Reports](#).

- You can schedule a report only in a connected client. For more information, see [How Siebel Runs Reports in Connected Clients](#).

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 [Customizing Siebel 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](#).

If you cannot access a report, then ask your administrator to assign the XMLP_SCHEDULER responsibility to you. For information, see [How Siebel Controls Access to Reports](#).
6. (Optional) Enter a custom name.

For information, see [Entering a Custom Name](#).
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](#).
9. View the report output.

For more information, see [Viewing Report Output](#).

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.

Value	Description
	◦ 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.
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 Run Time list, and then 30 in the Run Time list, then Siebel CRM runs the report at 11:30 PM according to the time zone that the Oracle Publisher Server uses. ◦ Start Date. Set the day, month, and year when Siebel CRM must run the first instance of this report. The Start Date and End Date use the date format that the user preference specifies. ◦ End Date. Set the day, month, and year when Siebel CRM must run the last instance of this report. ◦ Days. Choose the day or days of the week when Siebel CRM must run this report. ◦ Public. If you make this button active, then all users can view this report. ◦ Save Data. If you make this button active, then Siebel CRM saves the unformatted report data in an XML file. ◦ Save Output. If you make this button active, then Siebel CRM saves the formatted report output, such as a PDF file, and you can view this output in the Scheduled Reports view.
Run Monthly	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*.

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 Run Time list on the near side. Choose 30 in the Run Time list on the far side.
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.

Element	Value
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](#) and [Viewing Report Output](#).

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, then the Scheduled Reports view.
2. Monitor the status in Oracle Publisher by doing one of the following:
 - o 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.
 - o 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 Publisher repository. For more information, see [Deleting Scheduled Reports from Oracle Publisher Server](#).

4 Integrating Oracle Publisher with Siebel

Integrating Oracle Publisher with Siebel

This chapter provides instructions for administrators on how to integrate Oracle Publisher (BIP 11.1.1.9 or 12, OAP 5.5.0) with Siebel CRM for report generation, for first-time installations as well as upgrades. It includes the following topics:

- *About Integrating Oracle Publisher with Siebel*
- *Requirements for Using Siebel Reports for First-Time Installations*
- *Roadmap for Integrating Oracle Publisher with Siebel for First-Time Installations*
- *Preparing to Integrate Oracle Publisher with Siebel*
- *Process of Installing Oracle Publisher for Integration with Siebel*
- *Upgrading to the Latest Oracle Publisher Patch*
- *Configuring an Outbound Web Service for Siebel Reports*
- *Enabling Siebel Server Component Groups for Siebel Reports*
- *Configuring Security and Authentication for Siebel Reports*
- *Configuring Security and Authentication for Siebel Reports Using LDAP*
- *Configuring Siebel Reports in a Web Single Sign-On Environment*
- *Adding Explicit Reference to Container URL for Oracle Publisher Server*
- *Implementing EAI File Handling Changes for Siebel Reports Generation*
- *Uploading Preconfigured Report Layout Templates to Oracle Publisher Server*
- *Process of Enabling and Configuring Report Scheduling*
- *About Upgrading Siebel Reports for Siebel 17.0*

About Integrating Oracle Publisher with Siebel

Oracle Publisher (BIP or OAP) is the reporting module for Siebel CRM. Siebel integrates with Oracle Publisher for report execution and administration. Oracle WebLogic Server and Publisher Server are installed as part of Oracle Publisher installation.

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

Some Siebel Business Applications do not require a Siebel client. For example, Siebel Loyalty can use the Oracle Publisher libraries to run Oracle 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 Siebel Reports Architecture*.

Each new Siebel CRM release typically introduces new report features and functionality. However, before you can use these latest features for first-time installations, you must perform several postinstallation and configuration tasks. For a step-by-step view of these tasks, see *Roadmap for Integrating Oracle Publisher with Siebel for First-Time Installations*. If you are upgrading from previous releases, you only need perform the instructions in *About Upgrading Siebel Reports for Siebel 17.0*.

Requirements for Using Siebel Reports for First-Time Installations

This topic provides the requirements you must meet before you can use Siebel Reports for the first time. The following requirements must be met before using Siebel Reports:

- You must have the latest Siebel CRM version.

Some of the requirements include:

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

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

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

- Java Development Kit (JDK) with the JAVA_HOME system environment variable pointing to the JDK.
- Siebel Server components enabled:
 - Siebel Application Object Manager (AOM), for example, SCCObjMgr_enu for Siebel Call Center
 - EAI Object Manager
 - XMLP Report Server
 - (Optional) Workflow

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

- Oracle Publisher software
 - Oracle Business Intelligence Publisher (Oracle BIP) or Oracle Analytics Publisher (OAP)
 - Oracle Publisher Add-In for Microsoft Word (also known as Oracle Publisher Desktop)

Note: Platform, database, and other certifications for Siebel are found on My Oracle Support. For help searching for certifications, see *How to Check the Certification Matrix in the New My Oracle Support Site* (KB871388).

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

The following is a roadmap for integrating Oracle Publisher (BIP and OAP) with Siebel CRM.

The typical tasks you perform to integrate Oracle Publisher with Siebel CRM if you are implementing Siebel Reports for the first time are as follows:

1. *Preparing to Integrate Oracle Publisher with Siebel*
2. *Process of Installing Oracle Publisher for Integration with Siebel*
 - a. *Installing Oracle Publisher Enterprise*
 - b. *Installing Oracle Publisher Add-in for Microsoft Word*
 - c. *Copying JAR Files to Oracle Publisher Server*
 - d. *Enabling External File References on Oracle Publisher Server*
 - e. *Start/Stop Oracle Web Logic Server and Oracle Publisher*
 - f. *Upgrading to the Latest Oracle Publisher Patch*
3. *Configuring an Outbound Web Service for Siebel Reports*
4. *Enabling Siebel Server Component Groups for Siebel Reports*
5. *Configuring Security and Authentication for Siebel Reports*
 - o *Setting Up and Enabling the Siebel Security Model for Siebel Reports*
 - o *Setting Responsibilities in the Siebel Application*
 - o *Configuring Security and Authentication for Siebel Reports Using LDAP*
 - o *Configuring Siebel Reports in a Web Single Sign-On Environment*
6. *Adding Explicit Reference to Container URL for Oracle Publisher Server*
7. *Implementing EAI File Handling Changes for Siebel Reports Generation*
8. *Uploading Preconfigured Report Layout Templates to Oracle Publisher Server*
9. *Process of Enabling and Configuring Report Scheduling*
 - a. *Configuring the Data Service WSDL for Oracle Publisher Server*
 - b. *Verifying Symbolic URL Arguments that Embed Oracle Publisher Scheduling Views in Siebel Application*
10. *Test your configuration.*

Preparing to Integrate Oracle Publisher with Siebel

Oracle Publisher (BIP or OAP) is the reporting module for Siebel Reports. To generate Siebel Reports, you must install and configure both Siebel CRM and Oracle Publisher. This topic provides the necessary preparation tasks for a new deployment and supports the latest reports enhancements.

CAUTION: If you are upgrading from a previous Siebel Reports deployment, do not perform the steps described in this topic. Instead, see *About Upgrading Siebel Reports for Siebel 17.0*.

To prepare to integrate Oracle Publisher with Siebel CRM

1. Make sure you have met the requirements for Siebel Reports.

For the requirements, see *Requirements for Using Siebel Reports for First-Time Installations*.

2. Install the appropriate Java Development Kit (JDK) version and point the JAVA_HOME system environment variable to the JDK, if you have not already done so.
3. Before using Siebel Reports, make sure that the Siebel Server is running and the Siebel clients and Siebel Tools can connect to the server data source.

Process of Installing Oracle Publisher for Integration with Siebel

This topic describes how to install Oracle Publisher (BIP or OAP) as a first-time installation for integration with Siebel CRM.

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

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

1. *Installing Oracle Publisher Enterprise*
2. *Installing Oracle Publisher Add-in for Microsoft Word*
3. *Copying JAR Files to Oracle Publisher Server*
4. *Enabling External File References on Oracle Publisher Server*
5. (Optional) *Start/Stop Oracle Web Logic Server and Oracle Publisher*
6. *Upgrading to the Latest Oracle Publisher Patch*

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

Installing Oracle Publisher Enterprise

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

This task is a step in *Process of Installing Oracle Publisher for Integration with Siebel*.

To install Oracle Publisher Enterprise

1. Log in to Oracle Software Delivery Cloud available at <http://edelivery.oracle.com>.
2. Enter the export validation information, accept the license terms and export restrictions, and then click Continue.

3. Select Oracle Business Intelligence (for BIP) or Oracle Analytics Server (for OAP) as the product pack, and an operating system (for example, Microsoft Windows [32-bit]). Then click Go.
Note: You must select an operating system compatible with the computer on which Oracle Publisher Server is installed. For information about which platforms are supported, see Oracle® Business Intelligence Publisher Certification information at http://docs.oracle.com/cd/E12844_01/doc/bip.1013/e12692/toc.htm#sthref2 or Oracle Analytics Server (OAS) certification information at <https://docs.oracle.com/en/middleware/bi/analytics-server/administer-oas/appendix-certification-information.html>.
4. Select the appropriate Media Pack. For example, you might select Oracle Business Intelligence (11g) Media Pack for Microsoft Windows (32-bit) or Oracle Analytics Server 5.5 Media Pack.
5. Download the appropriate Oracle Publisher installer for your deployment.
Note: Typically administrators download and install Oracle Publisher Desktop (also known as Oracle Publisher Add-in for Microsoft Word) during Oracle Publisher Enterprise installation, however, you can install it any time. For information on installing this desktop software, see *Installing Oracle Publisher Add-in for Microsoft Word*.
6. Install Oracle Publisher by doing the following:
CAUTION: You must extract the .zip file to a directory whose name does not contain spaces. If the directory name contains spaces, the installation fails.
 - a. Decompress the installer into a directory of your choice.
 - b. Follow the instructions about installing Oracle Publisher on Oracle Application Server for Oracle WebLogic Server.
 - c. Verify the installation by making sure you have an Oracle - BIPHomeX/OAPHomeX entry in the Start Programs menu, where X is an incremental number depending on the number of Oracle installations you have.
7. Enable Superuser for Oracle Publisher Server by doing the following:
 - a. Start Oracle Publisher (for example, <http://BIPServerHostName:9777/xmlpserver>), and then log in to Oracle Publisher Server with administrator credentials.
Note: The URL for starting Oracle Publisher resides in the `BI_Publisher_readme.txt` file located in the Oracle Publisher installation folder. Administrator credentials are case sensitive.
 - b. Click the Admin tab, and then select Security Configuration (in the Security Center section).
 - c. Check the Enable Local Superuser check box, enter a Superuser name and password of your choice, and then click Apply.
After creating a local superuser, administrators can log in directly to Oracle Publisher Server when the Siebel Server is not running.

Installing Oracle Publisher Add-in for Microsoft Word

Use the following procedure to install Oracle Publisher Add-in for Microsoft Word (also known as Oracle Publisher Desktop) for integration with Siebel CRM and Siebel Reports.

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

This task is a step in *Process of Installing Oracle Publisher for Integration with Siebel*.

To install Oracle Publisher Add-in for Microsoft Word

1. Log in to Oracle Software Delivery Cloud available at: <http://edelivery.oracle.com>.

Note: This software is listed on Oracle Software Delivery Cloud as Oracle Publisher Desktop.

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

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

4. Select the appropriate product Pack and then click Continue. For example, you might select Oracle Business Intelligence (11.1.1.7) product pack for Microsoft Windows (32-bit) or OAS 5.5 Media Pack.
5. Download the appropriate Oracle Publisher installer for your deployment.

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

6. Install Oracle Publisher Add-in for Microsoft Word by doing the following:

- a. Decompress the appropriate .zip file according to the Media Pack you chose in Step 3.

CAUTION: You must extract the .zip file to a directory whose name does not contain spaces. If the directory name contains spaces, the installation fails.

- b. Run the installer.
- c. Verify the installation by making sure you have an Oracle Publisher Desktop application in the Start Programs menu.

Copying JAR Files to Oracle Publisher Server

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

This task is a step in *Process of Installing Oracle Publisher for Integration with Siebel*.

Before beginning this task, the administrator must create the Classes folder, as required, in one of the following locations:

```
// Oracle BI Publisher:  
<InstallLocation>\user_projects\domains\bifoundation_domain  
  
// Oracle Analytics Publisher:  
<InstallLocation>\user_projects\domains\bi\bidata\components\bipublisher\repository\Reports  
\SiebelCRMReports
```

To copy JAR files to Oracle Publisher Server

1. Copy the following files from the `SIEBSRV_ROOT\CLASSES` directory to one of the following locations, as required, where `<InstallLocation>` is the install path for Oracle Publisher.

```
// Oracle BI Publisher:
<InstallLocation>\user_projects\domains\bifoundation_domain\Classes

// Oracle Analytics Publisher:
<InstallLocation>\user_projects\domains\bi\bidata\components\bipublisher\repository\Reports
\SiebelCRMReports\Classes
```

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

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

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

Enabling External File References on Oracle Publisher Server

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

This task is a step in *Process of Installing Oracle Publisher for Integration with Siebel*.

To enable external file references on Oracle Publisher Server

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

For instructions, see *Start/Stop Oracle Web Logic Server and Oracle Publisher*.

Start/Stop Oracle WebLogic Server and Oracle Publisher

There are many instances where you must start, stop, or restart Oracle WebLogic Server and Oracle Publisher (BIP or OAP).

To start/stop Oracle Web Logic Server and Oracle Publisher

1. Stop the Oracle Publisher application.
2. Log in to Oracle Web Logic Server, click Deployments and then select Lock & Edit.
3. Navigate to the `bipublisher` application.
4. To stop an application:
 - Go to the Control tab and select **Force Stop now**.
 - Select the application you want to stop and then click Stop.
5. To start an application:
 - Go to the Control tab and select **servicing all requests**.
 - Select the application you want to start and then click Start.After the application starts, the State field shows a status of Active.

To start or stop OAP services

1. Navigate to the following directory:
`C:\Oracle_Home\user_projects\domains\bi\bitools\bin\`
2. Then execute one of the following commands, as required:
 - **Windows:** `start.cmd` Or `stop.cmd`.
 - **UNIX:** `./start.sh` Or `./stop.sh`.

Upgrading to the Latest Oracle Publisher Patch

To use the latest reporting features, it's recommended that you upgrade to the latest Oracle Publisher (BIP or OAP) release.

For more information about Oracle Publisher updates and information about installing Oracle Publisher updates in a Siebel environment, open a service request with My Oracle Support.

Configuring an Outbound Web Service for Siebel Reports

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

To configure an outbound Web service for Siebel Reports

1. Navigate to the Administration - Web Services screen, then the Outbound Web Services view.

2. In the Name field, query for PublicReportServiceService.
3. Change or verify that the service port address corresponds with Oracle Publisher Server.

The format is: `http://<WebServerHostName>:<port>/xmlpserver/services/ PublicReportService_v11`

where:

- *WebServerHostName* is the full path for Oracle Publisher Server
- *port* is the port on which Oracle Publisher Server resides. For example:

The address for BIP might be: `http://BIPServerHostName:7001/xmlpserver/services/
PublicReportService_v11`

The address for OAP might be: `http://OAPServerHostName:9502/xmlpserver/services/
PublicReportService_v11`

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

Enabling Siebel Server Component Groups for Siebel Reports

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

To enable server component groups for Siebel Reports

1. Navigate to the Administration - Server Configuration screen, Enterprises, then the Component Groups view.
2. Enable the following component groups on the Siebel Server:

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

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

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

Configuring Security and Authentication for Siebel Reports

Configure Security and Authentication for Siebel Reports

1. Review the following topics:
 - *About Security and Authentication for Siebel Reports*
 - *Reports Authentication in Siebel Reports*
 - *About Oracle Business Intelligence Enterprise Edition Coexistence with Siebel Reports*
2. Perform one of the following:
 - *Setting Up and Enabling the Siebel Security Model for Siebel Reports*
 - *Setting Responsibilities in the Siebel Application*
 - *Configuring Security and Authentication for Siebel Reports Using LDAP*
 - *Configuring Siebel Reports in a Web Single Sign-On Environment*

About Security and Authentication for Siebel Reports

It's recommended that you use the Siebel Security Model when configuring security and authentication for Siebel Reports. That model and a few others are discussed here.

Siebel Security Model

When Oracle Analytics Publisher receives a request from a Siebel user to perform report functionality on the server, it queries the Siebel application to retrieve the user privileges associated with that particular user. This retrieval is done by invoking a web service exposed by the Siebel application.

The Siebel Security model retrieves authentication information through the Siebel EAI Object Manager (EAIObjMgr) and, as a one-time event, authenticates sign-ins for report users against the Siebel database. After that, a session token is provided for Oracle Analytics Publisher authentication. Because every user executing reports has their own credentials and roles, users have private visibility on Oracle Analytics Publisher when scheduling reports.

Note: The reporting responsibilities in the Siebel application correspond to roles and permissions on Oracle Analytics Server (OAS). The names of these responsibilities must exactly match the corresponding set of roles on OAS.

LDAP

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

Web Single Sign-On Authentication

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

Oracle Analytics Publisher Security

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

Note: The Oracle Analytics Publisher security model doesn't enable private scheduled reports. All scheduled reports are considered public.

For more information, see the remaining topics in this chapter and the [Oracle Analytics Publisher](#) website.

About Oracle Business Intelligence Enterprise Edition Coexistence with Siebel Reports

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

- LDAP security model (recommended)

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

- Oracle BI Server Security model

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

About Configuring Security and Authentication for Siebel Reports

Depending on the kind of authentication you've set up, the application checks that a user exists in the directory as follows

- For Siebel Security, checks by way of the endpoint that a user exists in the Siebel application.
- For LDAP, checks for a user in the directory.

- For Oracle Analytics Publisher security, there's no user directory check. The generic authentication credentials are checked against a user defined in Oracle Publisher. In this authentication model, responsibilities aren't checked. Instead, the same generic user account is used for the connection to Oracle Analytics Publisher Server.

Note: For more information on Oracle Analytics Publisher Security, see *Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher* or *Administering Oracle Analytics Publisher in Oracle Analytics Server*.

- Checks that a user has certain roles, depending on what action they're taking:
 - For LDAP, groups are created that relate to Oracle Analytics Publisher roles. Users in the LDAP directory have these certain groups assigned to them. Oracle Analytics Publisher authentication to LDAP directory checks for a user if they have a certain group.
 - For Siebel Security, each user has certain XMLP responsibilities. Oracle Analytics Publisher authentication to the Siebel application, by way of an endpoint inbound web service call, checks that a user has the correct responsibilities that relate to the Oracle Analytics Publisher roles.
- Checks Siebel XMLP responsibilities that control the Siebel user interface for Reports menu restrictions (for example, XMLP_SCHEDULER responsibility for the scheduling option).

Setting Up and Enabling the Siebel Security Model for Siebel Reports

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

This task is a step in *Configuring Security and Authentication for Siebel Reports*.

To set up an inbound Web Service and enable Siebel Security Model for Siebel Reports

- Log in to the Siebel application with administrative privileges.
- Navigate to the Administration - Web Services screen, then the Inbound Web Services view.
- In the Name field in the Inbound Web Services list, query for BIPDataService and BIPSiebelSecurity.
- Add another Web service:

```
<-- BIPSiebelSecurity -->  
https://<server>:<port>/siebel/app/eai/Lang?SWEExtSource=SecureWebService&SWEExtCmd=Execute&WSSOAP=1
```

where:

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

For example:

```
https://<server>:<port>/siebel/app/eai/enu?SWEExtSource=SecureWebService&SWEExtCmd=Execute&WSSOAP=1
```

- Save and then click Clear Cache.

6. Enable the Siebel Security Model by doing the following:

- a. Make sure Oracle Publisher is started.
- b. Log in to the Oracle Publisher Enterprise application with administrator privilege.
- c. Click the Admin tab, and then select Security Configuration (under Security Center).
- d. In the Security Model section, perform the following, and then click Apply:

- Set the model to Siebel Security
- Enter the host name and port as the Web Service Endpoint URL

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

- Enter the Siebel administrator credentials

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

7. Restart Oracle Publisher Server.

For information about restarting Oracle Publisher Server, see *Start/Stop Oracle Web Logic Server and Oracle Publisher*.

8. Verify that the Siebel Security Model has been implemented by doing the following:

- a. Start the Oracle Publisher Enterprise application (for example, `http://<BIPServerHostName>:<port>/<xmlpserver>`).
- b. Log in to the Oracle Publisher Enterprise application using the Siebel administrator credentials.

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

Setting Responsibilities in the Siebel Application

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

This task is a step in *Configuring Security and Authentication for Siebel Reports*.

To set responsibilities in the Siebel application

1. Define seven new responsibilities in the Siebel application.

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

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

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

- All Report Output List View
- All Report Template Across Organization View
- All Report Template View
- Manager's Report Output List View
- Manager's Report Template View
- Report Job List View
- Report Output List View
- Report Standard Template Registration Admin View
- Report Template Registration Admin View

3. Associate the XMLP_SCHEDULER and XMLP_SCHEDULER_USER responsibility with the Report Job List View view.

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

Configuring Security and Authentication for Siebel Reports Using LDAP

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

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

This task is a step in *Configuring Security and Authentication for Siebel Reports*.

To configure security and authentication for Siebel Reports using LDAP

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

Configure Siebel Security for Oracle BI Publisher with LDAP/ADSI Security Adapter

This topic instructs you how to configure Siebel CRM 8.1.1.1 or higher and Oracle BI Publisher 10g and 11g to use the Siebel security model, when Siebel CRM is using LDAP/ ADSI Security Adapter for authenticating the Siebel users.

Siebel CRM Configuration

Here are the high-level steps to configure Siebel CRM:

1. Set up LDAP/ADSI authentication for the Siebel Application Object Manager that you're using, for example, Siebel Call Center or Siebel ePharma. It's also required that you first test that the LDAP/ADSI authentication works for the Siebel application.
2. Enable LDAP/ADSI authentication for both EAI Object Manager and XMLP report server components. Change the component parameters shown in the following tables, for both EAI Object Manager and XMLP Report Server components:

LDAP Authentication

Parameter Name	Value
Security Adapter Mode	LDAP
Security Adapter Name	LDAPSecAdpt
Username	<LDAPUSERID>
Password	User defined

ADSI Authentication

Parameter Name	Value
Security Adapter Mode	ADSI
Security Adapter Name	ADSIAdpt
Username	<ADSIUSERID>
Password	User defined

CAUTION: The user ID and password should be of a Siebel employee record that also has an LDAP/ADSI account. You can use the Anonymous User Account that's specified in the eapps.cfg file on the Siebel web server. It's REQUIRED to have all three components (for example, Application Object Manager, EAI Object Manager, and XMLP Report Server) use the same authentication store. For example, if Siebel Application Object Manager is using LDAP, then the EAI Object Manager and the XMLP Report Server component should also be configured to use LDAP, otherwise the Siebel - BI Publisher Reports integration won't work.

Oracle BI Publisher Configuration

Here are the high-level steps to configure Oracle BI Publisher:

1. Enable Siebel Security Model as documented in the Siebel Reports Guide > Overview of Siebel Reports > Integrating Oracle BI Publisher with Siebel Business Applications
For more information, see: [Integrating Oracle BI Publisher with Siebel CRM](#) (KB609727).
2. Ensure that the Administrator Username is set to the LDAP/ADSI user name. Also set the Administrator Password accordingly.

Once you complete the Siebel and BI Publisher configuration steps, restart the Siebel Server, Siebel Gateway, and Oracle BI Publisher services. Then, test the Report Template upload and eventually run the Siebel BI Publisher reports.

Example of Configuring Security and Authentication for Siebel Reports Using LDAP

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

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

To configure security and authentication for Siebel Reports using LDAP

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

4. Associate the LDAP users to relevant members.

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

Configuring Siebel Reports in a Web Single Sign-On Environment

This task is a step in *Configuring Security and Authentication for Siebel Reports*.

To configure Siebel CRM and Oracle Publisher for Web Single Sign-On (SSO) authentication, complete the following tasks:

- *Configuring Oracle Publisher to Integrate with Siebel for Web Single Sign-On*
- *Configuring Siebel to Integrate with Oracle Publisher for Web Single Sign-On*
- *Enabling Reports Scheduling with Web Single Sign-On*
- *Enabling TLS for Oracle Publisher Running on Oracle WebLogic Server*

Configuring Oracle Publisher to Integrate with Siebel for Web Single Sign-On

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

To configure Oracle Publisher to integrate with Siebel in Web SSO environment

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

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

Configuring Siebel to Integrate with Oracle Publisher for Web Single Sign-On

The following procedure shows how to configure your Siebel application so that it can integrate with Oracle Publisher when Web Single Sign-On (SSO) authentication is being used.

To configure Siebel to integrate with Oracle Publisher in Web SSO environment

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

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

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

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

Server Component	Parameter	Value

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

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

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

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

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

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

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

Enabling TLS for Oracle Publisher Running on Oracle WebLogic Server

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

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

To enable TLS for Oracle Publisher running on Oracle Web Logic server

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

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

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

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

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

```
Dweblogic.security.SSL.protocolVersion=TLSv1.2
```

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

Enabling Reports Scheduling with Web Single Sign-On

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

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

To enable Siebel Reports scheduling when Web SSO is implemented

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

For information about creating custom Siebel Server component definitions, see *Siebel System Administration Guide*.
2. Create a new Siebel enterprise profile (named subsystem) by copying the security adapter profile used by the Application Object Manager. Do the following:
 - o If the Siebel application is using the ADSISecAdpt security adapter profile, create a copy of the profile and name it ADSISecAdpt_NoSSO.

For information about creating Siebel Enterprise Server named subsystems, see *Siebel System Administration Guide*.
3. Set the SSO profile parameter for the new security adapter profile you created in Step 2 to False.

4. For the BIP EAI Object Manager component you created in Step 1, specify values for the parameters shown in the following tables:

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

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

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

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

- a. Add the following basic information parameters:

```
Application Name = bipeai
Language = <lang>
Object Manager = EAI Object Manager(ENU)
Configure EAI HTTP Inbound Transport = TRUE
```

- b. Add the following enhanced authentication parameters:

```
Anonymous User Name : <Guest Login>
Anonymous User Password: <Guest Password>
```

7. Create a second application as part of the Application Interface profile and do the following:

- a. Add the following basic information parameters:

```
Application Name = eai_anon
Language = <lang>
Object Manager = EAI Object Manager(ENU)
Configure EAI HTTP Inbound Transport = TRUE
Configure Anonymous Pool = TRUE
Anonymous Pool Size = <Pool Size>
```

- b. Add the following enhanced authentication parameters:

```
Anonymous User Name : <Guest Login>
Anonymous User Password: <Guest Password>
```

8. Launch the Siebel Web Client and log into the Siebel application as a Siebel administrator.
9. Navigate to the Administration - Web Services screen, then the Inbound Web Services view.
10. In the Name field of the Inbound Web Services list, query for BIPDataService.
11. In the address URL for the BIPDataService, change the value `eai_lang` to `eai_anon_lang`. For example:

```
http://SiebelWebServerName/eai_anon_lang/
start.swe?SWEExtSource=WebService&SWEExtCmd=Execute&WSSOAP=1
```

12. Click the Generate WSDL button to generate a WSDL file, then save the file with the name `dataservice.wsdl`.
13. Copy the `dataservice.wsdl` file to the Oracle Publisher home directory. By default, this is the `<root_dir>\user_projects\domains\bifoundation_domain` directory on Oracle Publisher server.
14. Restart the Publisher WebLogic server.

Adding Explicit Reference to Container URL for Oracle Publisher Server

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

To add a container URL for Oracle Publisher Server

1. Navigate to the Administration - Server Configuration screen, Enterprises, and then the Profile Configuration view.
2. Select the XMLPJvmSubsys profile.
3. In the Profile Parameters subview, set the value of the Container URL field as follows:
`http://<serverName>:<Port>/siebel/jbs`
where:
`Port` is the port number used for the config agent port parameter.
4. Restart the Siebel Services.

Implementing EAI File Handling Changes for Siebel Reports Generation

Siebel Fix Packs 8.1.1.11 and 8.2.2.4 introduced the `EAIFileTransportFolders` enterprise parameter. This parameter lets you restrict write access for the EAI File Transport to specific folders within the Siebel file system. However, for report generation to work properly, you must change the value of this parameter specific to your deployment.

For more information about this new feature, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

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

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

To implement EAI file handling changes for Siebel Reports generation

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

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

Note: The SIEBSRVR_ROOT\ses\siebsrvr\xmlp\data directory is for report scheduling and the directory is available by default.

4. Restart the Siebel Server.
5. Do this only for Siebel Mobile Web Client and Siebel Developer Client:
 - a. Add a new section in the application.cfg file as follows:

```
[EAIFileTransportConfigSubsys]
EAIFileTransportFolders = c:\SIEBSRVR_ROOT\ses\siebsrvr\xmlp\reports;
c:\SIEBSRVR_ROOT\ses\siebsrvr\xmlp\data
```

- b. Update the following section in the application file as follows:

```
[XMLPJvmSubsys]
FullName = XMLPJvmSubsys
Description = XMLP Java Business Service Subsystem Parameters
SubsysType = JavaContainerSubSys
CONTAINERURL = http://<ServerName>:<port>/siebel/jbs
```

Where: `port` is the port number used for the config agent port parameter.

Related Topics

- [Troubleshooting Error Messages for Siebel Reports](#)

Uploading Preconfigured Report Layout Templates to Oracle Publisher Server

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

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

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

To upload preconfigured report layout templates to Oracle Publisher Server

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

2. From the application-level menu, select Edit, and then Select All.
Note: If you receive a SBL-RPT-50582 error message, upload the preconfigured reports individually. This error means that a valid XLIFF or ZIP file has not been associated with the report.
3. Click Upload Files.
Note: When you create your own custom reports, you must upload the report template and associate it with a view before you can use that report.
4. Navigate to the Administration - BI Publisher Reports screen, then the View Association view to associate the Account List report to the Account List view.
The Account List report layout template is now available to the Siebel application for use in generating reports.
5. Verify Oracle Publisher integration by logging in to Oracle Publisher Server with administrative privilege.
6. Test report generation by performing the following:
 - a. Navigate to the Accounts screen, then the Accounts List view.
 - b. Perform a query that returns approximately 30 records.
 - c. In the application toolbar, click the Reports button.
 - d. In the Publisher section, choose Account List.
 - e. (Optional) Select the report output type, and then click Submit.**Note:** If you receive error message SBL-RPT-50524 or SBL-EAI-50228, even though you have applied Siebel CRM Fix Pack 8.1.1.11 (or 8.2.2.4), see 1542583.1 (Article ID) on My Oracle Support for resolution – note that this is specific to Oracle BIP; it does not apply to OAP.

Process of Enabling and Configuring Report Scheduling

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

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

1. *Configuring the Data Service WSDL for Oracle Publisher Server*
2. *Verifying Symbolic URL Arguments that Embed Oracle Publisher Scheduling Views in Siebel Application*

Configuring the Data Service WSDL for Oracle Publisher Server

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

This task is a step in *Process of Enabling and Configuring Report Scheduling*.

To configure the data service WSDL for Oracle Publisher Server

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

```
https://<SiebelServer>:<Port>/siebel/app/eai_anon/<Lang>?  
SWEExtSource=SecureWebService&SWEExtCmd=Execute
```

where:

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

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

5. Select BIPDataService, and then click Generate WSDL.
6. Save the WSDL as a new data service file named `dataservice.wsdl` in a directory in the Oracle Publisher home directory (by default, this is the `<InstallLocation>\user_projects\domains\bifoundation_domain` directory). For example, you might save the file as:

```
<InstallLocation>\user_projects\domains\bifoundation_domain\dataservice.wsdl
```

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

The WSDL is now available for importing.

Verifying Symbolic URL Arguments that Embed Oracle Publisher Scheduling Views in Siebel Application

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

This task is a step in *Process of Enabling and Configuring Report Scheduling*.

Verify Symbolic URL Arguments

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

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

5. Add new records to Symbolic URL Arguments with the values shown in the following table.

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

6. Restart the Siebel Server.

About Upgrading Siebel Reports for Siebel 17.0

If you are currently using Siebel Reports and you plan to upgrade to Siebel CRM 17.0, then you must do the following:

- Upgrade your Siebel application (or applications).

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

- Upgrade to the latest Oracle Publisher release.

For information on upgrading to the latest Oracle Publisher release, see *Upgrading to the Latest Oracle Publisher Patch*.

5 Administering Siebel Reports

Administering Siebel Reports

This chapter describes how to administer Siebel Reports. It includes the following topics:

- *Doing Typical Administrative Tasks*
- *Optimizing the Performance of Siebel Reports*
- *Administering Multilingual Reports*
- *Administering Signatures in Reports*

Doing Typical Administrative Tasks

Some typical administrative tasks include the following:

- *Disabling the Reports Button*
- *Making Sure Siebel Displays The Report Status*
- *Making Sure Fonts Are Available for Reports*
- *Managing Report Notifications*
- *Viewing Report Usage Statistics*
- *Configuring Siebel to Automatically Delete Reports*

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 *Siebel Reports Business Service*.

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 Displays The Report Status

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

To make sure Siebel displays the report status

- Enable the XMLP Report server component.

Making Sure Fonts Are Available for Reports

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

To make sure fonts are available for reports

1. Log on to the Siebel Server where you enabled the XMLP Report server component.
2. Copy the fonts, depending on the following operating system that your deployment uses:
 - **Windows.** Copy the fonts that the following folder contains to the fonts subfolder that resides in the Java installation folder for your environment. This subfolder is typically `JAVA_HOME\lib\fonts`:

`C:\WINDOWS\Fonts`
 - **UNIX.** Copy the fonts that reside in the `/Fonts` folder to the fonts subfolder in the Java installation folder for your environment. For example:
 - **AIX.** `/usr/lpp/X11/lib/X11/fonts/TrueType`
 - **Linux.** `/usr/lib/X11/fonts/TrueType`

Managing Report Notifications

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

- If the user clicks Submit, then Siebel CRM immediately returns to the Siebel application. The user is not required to wait for the report to finish running. The user can navigate to another view or do some other work while the report runs in the background.
- Displays a notification in the report icon when the report finishes running.

- Allows the user to view report notifications and to open a completed report without navigating to the My Reports view.
- Hides the notification when the user opens the report.
- Displays a report notification only to the user who runs the report, and only for the current user session. It does not display the notification to any other user. It does not display the notification if the user logs out and then logs back in to the client.

To disable reports notification

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

Viewing Report Usage Statistics

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

Configuring Siebel to Automatically Delete Reports

This topic describes how to configure Siebel to automatically delete reports, from Siebel File System and Oracle 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 Siebel File System or Oracle 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* and *Types of Users Who Use Siebel Reports*.

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 Oracle Publisher Server

Siebel CRM stores scheduled reports that users create in the Oracle Publisher scheduling database. It is recommended that you configure Siebel CRM to regularly delete these reports.

To remove scheduled reports from Oracle Publisher Server

- Use the Job Manager Instances List or the Job Manager Purge Instances Window to remove Scheduler job instances from the scheduling database (in BIP) or from the catalog page (in OAP).

For more information, see the topic about Purging Oracle 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*
- *Setting the Polling Interval*
- *Setting Concurrency Parameters*
- *Optimizing Reports That Include Thousands of Records*

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*
- Running reports, see 1392449.1 (Article ID), Siebel BI Publisher Report Generation, on My Oracle Support

Setting the Waiting Period

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

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

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

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

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

To set the waiting period

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

Setting the Polling Interval

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

To set the polling interval

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

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

6. Restart the Siebel Server.

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

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

Setting Concurrency Parameters

It is recommended that you set the concurrency parameters.

To set concurrency parameters

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

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

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 you schedule this report to run on the Oracle Publisher Server. It is also recommended that you run this report immediately. For more information, see [Scheduling Reports](#).

Modifying the HTTP Sleep Time

The integration from Siebel CRM to Oracle Publisher uses an outbound web service call that calls the PublicReportService Web service on the Oracle 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 don't schedule this report, then the report times out and Siebel CRM adds an entry that's 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.

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

7. Restart the server.

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

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

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

1. Log in to Siebel client with administrative privileges.
2. Navigate to the Administration - Business Service screen, then the Simulator view.
3. In the Simulator list, click New, and then set the fields with the values shown in 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* and is applicable with a connected client.

To set server profile parameters

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

Enabling Scalable Mode

This topic describes how to set the scalable mode for all reports. Setting the scalable mode prepares the 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 Publisher-FO Processor, see *About How Siebel Runs Reports in Clients*.

To enable scalable mode

1. Log on to the Oracle Publisher Server.
2. Navigate to the following folder:

```
\BI\user_projects\domains\bifoundation_domain\config\bipublisher\repository\Admin\Configuration
```

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

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

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

6. Save the xdo.cfg file.

Configuring a Temporary Folder

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

To configure a temporary folder

1. Complete Step 1 through Step 4 of *Enabling Scalable Mode*.
2. Configure a temporary folder on Oracle 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 Oracle 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 WebLogic process might consume too much Java Virtual Memory. To avoid a reports failure, you can increase the allocation that the Java Virtual Memory uses for the WebLogic process.

To increase memory for the Java Virtual Machine

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

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

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

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

Administering Multilingual Reports

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

- [How Siebel Runs Multilingual Reports](#)
- [Creating Multilingual Reports](#)

How Siebel 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.

- Installing Siebel Language Packs, see *Siebel Installation Guide*.
- Deploying multiple languages, see *Siebel Global Deployment Guide*.

How Siebel 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 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*.

Priority That Determines the Report Language

The following table describes the priority that Siebel CRM uses to determine the report 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 <i>Running Reports</i> .
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 <i>Creating Multilingual Reports</i> .
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 <i>Siebel Reports Business Service</i> .
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 <i>Setting User Preferences for Siebel Reports</i> .
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's 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 need.

For more information about externalizing the user interface elements, see *How Siebel Translates User Interface Elements*. For more information about designing reports, see *Customizing Siebel 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*.

Create a Multilingual Report

1. Navigate to the Administration - Data screen, and then the List of Values view.
2. Query for XMLP_RPT_OUTPUT_TYPE and set the Display Value setting to ALL.
3. Select these check boxes:
 - Active
 - Translate
 - Multilingual
4. (Optional) Repeat steps 2 and 3 for each output type you want.
5. Design a report in the default language.

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

6. Use Oracle 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 *Oracle Publisher Desktop*.

7. (Optional) Use Oracle Publisher Desktop to add a default locale to the report template:
 - a. In Oracle Publisher Desktop, open the report template.
 - b. In the Publisher menu, select Tools, select Translate Template, and then select Localize Template.

Do the work described in *Registering Report Templates*, with these differences:

- o Use this format when you name the 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.
- The extension is XLF, RTF, or XLS.

For example, if the template name is EmployeeTemplate and if you must upload a Japanese-Japan translation, then use this file name: `EmployeeTemplate_ja_JP.xlf`.

- o Use this format when you name the ZIP file that you upload. File names aren't case-sensitive: `report template file name.zip`.
- o 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 can't 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 this work according to the type of file you upload:

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

For more information about this directory structure, see *Directory Structure That Siebel Reports Uses*.

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 the section about validation in *How Siebel Uploads Report Templates*.

- o When you validate the directory structure of Oracle Publisher Server, note that Siebel CRM uses this format to register each XLIFF file: `TemplateName_language code_TERRITORY CODE.xlf`.

For more information about using this format, see *Work with Translation Files* in Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server on *Oracle Help Center/Middleware/BI*.

Administering Signatures in Reports

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

- [Creating and Modifying an Integration Object for Signatures in Reports](#)
- [Customizing a Report Template for Signatures](#)

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

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

For more information, see [Copying JAR Files to Oracle Publisher Server](#).

Creating and Modifying an Integration Object for Signatures in Reports

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

To create and modify an integration object for signatures in reports

1. Create a new integration object.

For more information, see [Creating New Integration Objects to Add Fields to Reports](#).

2. In the Object Explorer, expand the Integration Object tree, expand the Integration Component tree, and then click Integration Component Field.
3. Scroll through the Integration Component Field list until you locate the Signature field.
4. Confirm the following Signature field properties:
 - **XML Tag.** ssSignature
 - **ID.** ssId

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

Customizing a Report Template for Signatures

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

Note: For more information on creating report templates, see [Creating Report Templates](#).

To customize a report template for signatures

1. Open Oracle Publisher Desktop.

For more information, see *Oracle Publisher Desktop*.

2. Select and open the OOTB LS_SIGNAD.rtf template in the following folder: `SIEBEL_SERVER_ROOT\XMLP\TEMPLATES`.
3. In the OOTB LS_SIGNAD.rtf template, double-click the fields and rename them as necessary.

For example, if you want to use this report to obtain signatures for an invoice, then you can replace `<?for-each:ssPharmaCallSignature?>` with `<?for-each:ssFsInvoiceSignature?>`.

4. In the `ootb LS_SIGNAD.rtf` template, remove the following fields:
 - `ssContactLast`
 - `ssContactFir`
 - `concat(concat(ss`
5. Save the report template in RTF format.

6 Siebel Reports Architecture

Siebel Reports Architecture

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

- *About Siebel Reports Architecture*
- *About How Siebel Runs Reports in Clients*
- *How Siebel Reports Uses Integration Objects*
- *Computer Languages That Siebel Reports Uses*
- *Siebel Reports Business Service*
- *Directory Structure That Siebel Reports Uses*
- *Directory Structure That Oracle Publisher Repository Uses*
- *How Siebel Controls Access to Reports*
- *How Siebel Uses Report Templates With Organizations*

About Siebel Reports Architecture

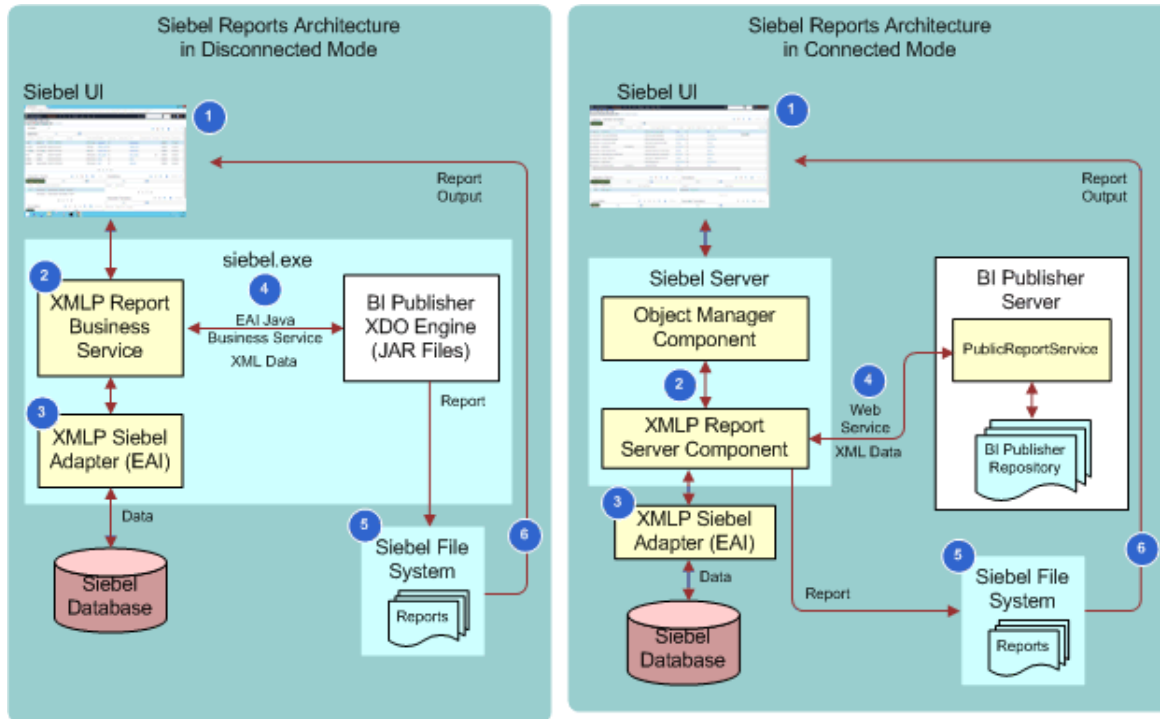
Siebel CRM uses the following separate architectures starting with Siebel CRM versions 8.1.1.x and 8.2.2.x:

- **Connected architecture.** Siebel CRM integrates directly with the Siebel XMLP Report server component and Oracle Publisher Server. For more information, see *How Siebel Runs Reports in Connected Clients*.
- **Disconnected architecture.** Siebel CRM uses Oracle Publisher to run reports through the XMLP Report Business Service. For more information, see *How Siebel Runs Reports in Disconnected Clients*.

About How Siebel Runs Reports in Clients describes how Siebel CRM typically runs reports for both connected and disconnected clients.

About How Siebel Runs Reports in Clients

The following figure illustrates how Siebel CRM typically runs reports in clients (connected and disconnected).



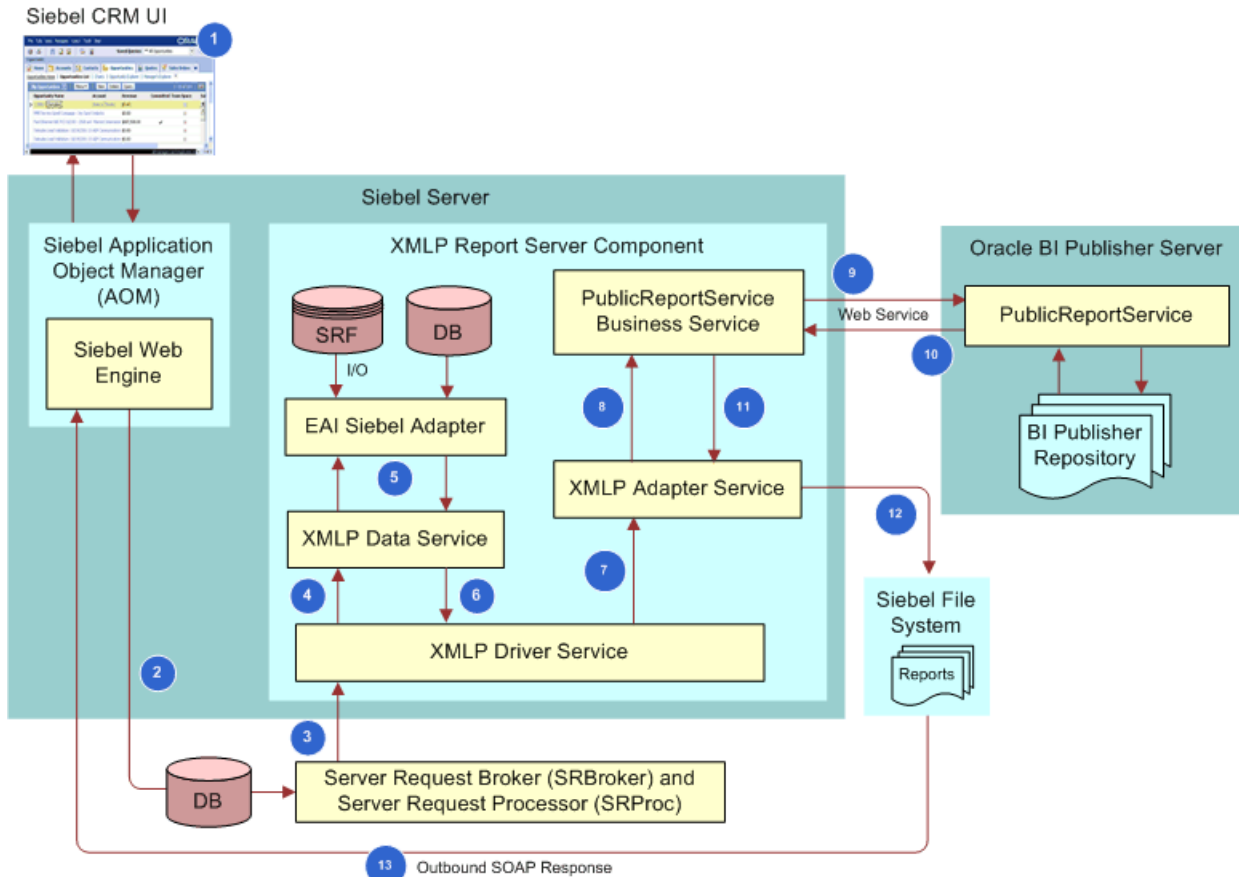
As shown in this figure, the typical steps involved when Siebel CRM runs a report for connected or disconnected clients are:

1. When a user requests to run a report in the Siebel client, Siebel CRM sends the request to one of the following:
 - Oracle Publisher Server, if the user is using a connected client
 - Oracle 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 Publisher-FO Processor.
3. The 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*.

How Siebel Runs Reports in Connected Clients

The following figure illustrates the architecture that Siebel CRM uses to run a report if the user is using a Siebel client that is connected to the Siebel Server.



As shown in this figure, the typical steps involved when Siebel CRM runs a report for connected clients are:

Note: As of Siebel CRM 17.0, the Siebel Repository File (SRF) no longer applies and is replaced by the Siebel Runtime Repository in Siebel database.

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 Oracle Publisher Server.

9. The PublicReportService business service makes a Web service call to Oracle Publisher Server. For more information, see *How Siebel Uses the PublicReportServiceService Web Service*.

10. Oracle Publisher Server runs the report, and then makes a Web service call to the PublicReportServiceService business service to return the report binary data.

Oracle 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 Publisher repository. It then sends the report to the Siebel application. For more information, see *How Siebel Uses Oracle Publisher Repository*.

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 Uses the PublicReportServiceService Web Service

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

- Validates privileges,
- Obtains information about reports, the Oracle Publisher repository, and the Oracle Publisher Server `SharedFolder/SIEBELCRMREPORTS`.
- Runs reports,
- Creates and manages reports,

How Siebel Uses Oracle Publisher Repository

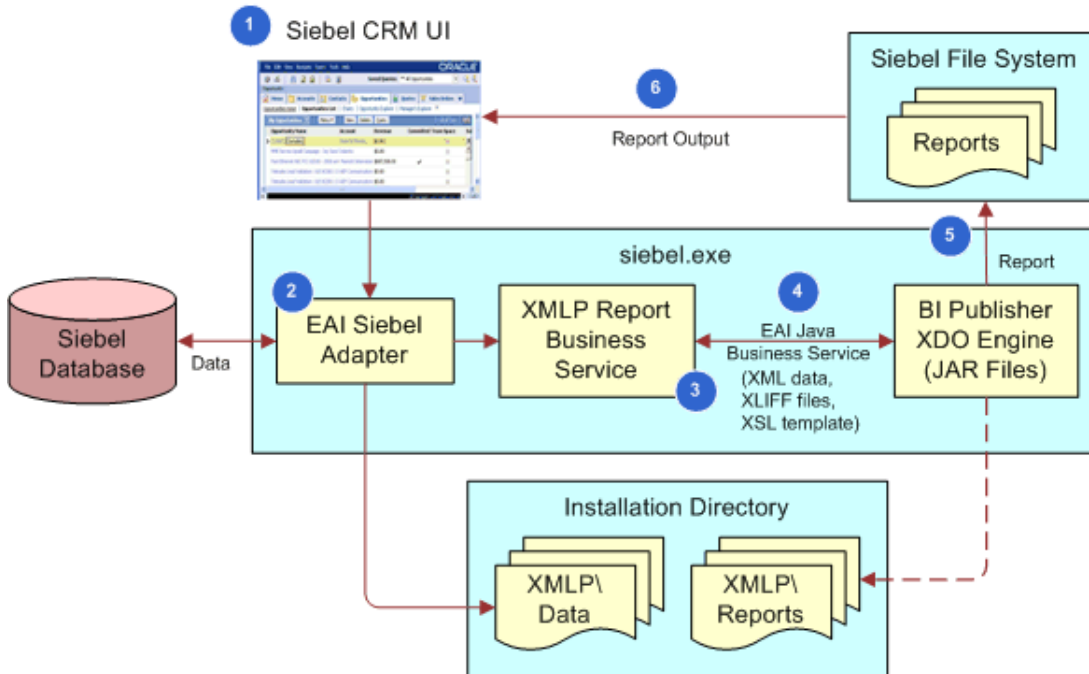
Oracle Publisher repository contains predefined reports and new reports that you create. It resides on Oracle Publisher Server. You install it when you install Oracle Publisher Server. The following folder in Oracle Publisher repository stores all reports:

`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*.

How Siebel Runs Reports in Disconnected Clients

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



As shown in this figure, the typical steps involved when Siebel CRM runs a report for disconnected clients are:

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 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 Publisher uses the EAI Java Business Service to send the XML file to the 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 Oracle Publisher displays the report in the client after storing the file in the Siebel File System.

The XMLP Report Business Service and Oracle 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*.

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 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 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 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 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](#) and [Creating New Integration Objects to Add Fields to Reports](#).

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 Runs Reports in Disconnected Clients](#).

- **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*.

Directory Structure That Siebel Reports Uses

The following table describes the folders that Siebel Reports uses. These folders reside in the following folder on the Siebel Server: `SIEBEL_SERVER_ROOT\XMLP`.

Directory	Description
<code>SIEBEL_SERVER_ROOT\XMLP\DATA</code>	Contains the XML files that Siebel CRM creates from integration objects. For more information, see <i>How Siebel Reports Uses Integration Objects</i> .
<code>SIEBEL_SERVER_ROOT\XMLP\FONTS</code>	Contains the predefined .ttf (TrueType format) font files. For more information, see <i>Making Sure Fonts Are Available for Reports</i> .
<code>SIEBEL_SERVER_ROOT\XMLP\REPORTS</code>	Contains the temporary files that Siebel CRM uses to run a report.
<code>SIEBEL_SERVER_ROOT\XMLP\TEMPLATES</code>	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 <i>About Siebel Reports Architecture</i> .
<code>SIEBEL_SERVER_ROOT\XMLP\xliff\language_code</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 <code>xliff</code> 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 <code>xliff\deu</code> folder. • XLIFF files for French reside in <code>xliff\fra</code> folder. Siebel CRM registers a report template, and then stores the XLIFF file in an <code>SIEBEL_SERVER_ROOT\XMLP\language_code</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 Oracle Publisher Server when you upload the report template.

Directory Structure That Oracle Publisher Repository Uses

The following table describes example folders that Oracle Publisher Repository uses with a connected client.

Example Directory	Description
<code>ORACLE_HOME/xmlp/XMLP/Reports/SiebelCRMReports/Account List/</code>	This directory contains the following files: • <code>aclist.xdo</code> file. Contains the report definitions. • <code>aclist.rtf</code> file. Contains the report template. • <code>aclist_ja_JP.xlf</code> file. Example file of a language translation. • <code>aclist_de_DE.xlf</code> file. Example file of a language translation.
<code>ORACLE_HOME/xmlp/XMLP/Reports/SiebelCRMReports/Contact List/</code>	This directory contains the following files: • <code>cnlist.xdo</code> file. Contains the report definitions. • <code>cnlist.rtf</code> file. Contains the report template. • <code>cnlist_ja_JP.xlf</code> file. Example file of a language translation. • <code>cnlist_de_DE.xlf</code> file. Example file of a language translation.

How Siebel 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 *Integrating Oracle Publisher with Siebel*.
- User access in relation to Siebel Reports, see *Integrating Oracle Publisher with Siebel*.
- Granting access to other users, see *Sharing Report Templates*.
- General information about controlling access, see *Siebel Security Guide*.

How Siebel Uses Report Templates With Organizations

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

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

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

7 Siebel Reports Business Service

Siebel Reports Business Service

You can use the Siebel Reports Business Service to run, share, or print a report automatically without any user interaction. You can 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 Reports Business Service method.

For example, you can create a workflow process that automatically:

- 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

Note: The Reports Business Service can't be run on the Mobile web client.

The reports are generated in the background by connecting to the Oracle Analytics Publisher server. The Siebel file system stores the report output, which users access from the My BI Publisher Reports view.

Note: BI Publisher has been rebranded as Oracle Analytics Publisher.

Guidelines for Modifying the Reports Business Service

If you need to modify the Report Business Service, then use these guidelines:

- Make sure you're familiar with the following skill set:
 - Using Siebel Tools
 - Using the Siebel Business Process Designer
 - Using Oracle Analytics Publisher Server
 - Modifying a business service
 - Scripting
- Make sure Oracle Analytics Publisher is integrated. This integration lets Siebel CRM use the business service methods of the Report Business Service.
- If you use the Reports Business Service to upgrade from a prior release, then make sure you do the following work:
 - Make sure the following workflow processes are active:
 - Report Generation for Oracle Publisher
 - Create Report Output for Oracle Publisher

- You can use the language code as an optional input argument in the XMLP Driver Service business service.

Activate the Siebel BI Publisher 8 Reports Business Service: Quick Steps

Aimed at Siebel configurators, this topic describes how to activate the Siebel Reports Business Service quickly. The Siebel Reports Business Service service enables Oracle Analytics Publisher (formerly called Oracle BI Publisher) reports to be generated via a workflow process, through scripting, or a with a business service from the Siebel 8.1.1 application.

Setup Overview for the Reports Business Service

Here are the high-level steps to use the Reports Business Service and the example workflow process:

1. Add additional methods and arguments to the XMLP Business Services.
2. Add example workflow processes.
3. Add a new business object.
4. Compile the SRF.
5. Activate and deploy the workflow processes.

To use a longer, more detailed procedure, see [Activate the Siebel BI Publisher 8 Reports Business Service: Overview of the Complete Steps](#).

Quick Setup for the Reports Business Service

Use these SIF and XML files to import the objects detailed in this topic.

- BIP Business Service.sif
- Report Output Business Object.sif
- BIP Create Report Output WF Process.sif
- BIP Report Generation WF Process.sif
- BIP Workflow Simulator.xml
- BIP Workflow Simulator Input Args.xml

The files are included in the Reports Business Service.zip archive file.

Here are the steps:

1. Back up all XMLP* business services to a .sif archive file.
2. Import the .sif file, BIP Business Service.sif, which includes the changes and new methods for XMLP Business Services.
3. Import the .sif file, Report Output Business Object.sif, which will create a new business object.
4. Import the .sif files, BIP Create Report Output WF Process.sif and BIP Report Generation WF Process.sif, which will create example workflow processes.
5. Compile the SRF.
6. Sign back into the application.

7. Activate the new workflow processes in the application:
 - a. Go to Site Map > Administration - Business Process > Workflow Deployment > Active Workflow Processes.
 - b. Query for **BIP***.
 - c. Select and activate.

Note: Use the Merge option instead of override during .sif import.

8. Test the setup. See *Test the Siebel BI Publisher 8 Reports Business Service Setup*.

Test the Siebel BI Publisher 8 Reports Business Service Setup

1. Execute the workflow by navigating to Administration - Business Service > Simulator.
2. Create a new record and click Load File. Open the file **BIP Workflow Simulator.xml** to load the example workflow process. Or, simply enter these values:
Business Service: Workflow Process Manager
Method Name: RunProcess
3. Enter the input arguments. Click Load File. Open the file **BIP Workflow Simulator Input Args.xml**. This will load the input arguments shown in the following table. Change these according to your environment.

Activate the Siebel BI Publisher 8 Reports Business Service: Overview of the Complete Steps

Here are the high-level steps to activate the Siebel BI Publisher 8 Reports Business Service:

Step	Where To Find More Information
Step 1 Add methods to the business service in Siebel Tools	• <i>Add Arguments to the XMLP Data Service Method</i> • <i>Add Arguments to the XMLP Adapter Service Method</i> • <i>Add Arguments to the XMLP Driver Service Method</i>
Step 2 Create a business object	<i>Create a Business Object</i>
Step 3 Create a workflow process	<i>Create a Workflow Process</i>
Step 4	<i>Deploy and Test the Configuration</i>

Step	Where To Find More Information
Deploy and test the configuration	

Related Topics

- [Activate the Siebel BI Publisher 8 Reports Business Service: Quick Steps](#)

Add Arguments to the XMLP Data Service Method

In Siebel Tools, for the existing Oracle Analytics Publisher business services, arguments for some methods need to be added to support workflow configurations.

These XMLP Data Service methods are covered here:

- BuildMultiXMLData
- BuildXMLData
- GetDataAsBinary

BuildMultiXMLData

Description: This method is used for generating sample data for multiple integration objects.

Method Arguments: Add the following method arguments for the BuildMultiXMLData method:

Name	Data Type	Type	Optional
IntegrationObject	String	Input	False

Inputs

Here are the available inputs and values for the BuildMultiXMLData method:

Name	Value
IntegrationObject	BIP Accounts - Current Query
BIP Accounts - Current Query	IO
BIP Current Account Service Profile	IO

BuildXMLData

Description: This method is used for generating sample data for a single integration object.

Method Arguments: Add the following method arguments for the BuildXMLData method:

Name	Data Type	Type	Optional
IntegrationObject	String	Input	False

Inputs

Here are the available inputs and values for the BuildXMLData method:

Name	Value
IntegrationObject	BIP Accounts - Current Query

GetDataAsBinary

Description: This method is used for pushing the runtime Siebel data in binary form to the Oracle Analytics Publisher Server.

Method Arguments: Add the following method arguments for the GetDataAsBinary method:

Note: The method arguments in this table are all String data types.

Name	Description	Type	Optional	Storage Type
EnableChunk	An indicator that specifies chunking or non-chunking during the pushing of Siebel data and downloading of the report output between Siebel and the Oracle Analytics Publisher Server.	Output	False	Property
IntegrationObject	Name of the integration object for the primary business object.	Input	False	Property
Bookmark	The user-specified search specification on the active view. For example, [Name] Like '3Com*'	Input	True	Property
ViewMode	Visibility of the current active view. For example, Organization	Input	True	Property
reportFileName	The temporary file generated for the Siebel data in the Oracle Analytics Publisher Server.	Output	True	Property
<Value>	Siebel data for the report generation in binary form.	Output	True	Value

Inputs

Here are the available inputs and values for the GetDataAsBinary method:

Name	Value
IntegrationObject	BIP Accounts - Current Query
Bookmark	[Name] Like 'A*'
ViewMode	Organization

Outputs

Here are the available outputs and values for the GetDataAsBinary method:

Name	Value
EnableChunk	TRUE
reportFileName	xmlp91996317.tmp

Add Arguments to the XMLP Adapter Service Method

In Siebel Tools, for the existing Oracle Analytics Publisher business services, arguments for some methods need to be added to support workflow configurations.

This XMLP Adapter Service method is covered here: RunBIPReport.

Note: The method arguments in this table are all String data types.

RunBIPReport

Description: This method is used to generate the report.

Method Arguments: Add these method arguments for the RunBIPReport method:

Note: The method arguments in this table all have the following attributes:

- Type: Input
- Data Type: String
- Storage Type: Property

Arguments for the RunBIPReport Method

Name	Optional
BIPPassword	False

Name	Optional
BIPUserId	False
EnableChunk	False
LanguageCode	False
ReportName	False
ReportOutputId	False
ReportOutputType	False
ReportSelected	False
reportFileName	True

Inputs

Here are the available inputs and values for the RunBIPReport method:

Name	Value
BIPUserId	Administrator
BIPPassword	Administrator
EnableChunk	True
LanguageCode	ENU
ReportName	Account List
ReportOutputId	88-320FR
ReportOutputType	PDF
ReportSelected	Account List
reportFileName	xm1p91996317.tmp

Add Arguments to the XMLP Driver Service Method

In Siebel Tools, for the existing Oracle Analytics Publisher business services, arguments for some methods need to be added to support workflow configurations.

This XMLP Driver Service method is covered here: GenerateBIPReport.

GenerateBIPReport

Description: This method is used to run the report. This is the final method required for report generation. This method internally generates Siebel binary data and pushes it to the BI Publisher server.

Method Arguments: Add these method arguments for the GenerateBIPReport method:

Note: The method arguments in this table all have the following attributes:

- Type: Input
- Data Type: String
- Storage Type: Property

Arguments for the GenerateBIPReport Method

Name	Description	Optional
BIPPassword	The BI Publisher server password for session authentication.	False
BIPUserId	The BI Publisher server User ID for session authentication.	False
EnableChunk	An indicator that specifies chunking or non-chunking during report download output between Siebel and BI Publisher Server.	False
LanguageCode	Language code where Siebel Object Manager is running.	False
ReportName	Language-independent report name to be generated.	False
ReportOutputId	Row ID of Report Output BC for the generated report.	False
ReportOutputType	The output format the report must be generated in.	False
ReportSelected	The language-dependent report name.	False
reportFileName	The temporary file generated for the Siebel data in the BI Publisher server.	True

Reports Business Service Methods You Can Use with the XMLP Driver Service

This table describes the methods that you can use with the XMLP Driver Service of the Reports Business Service:

Note: Methods used **internally** in Siebel's Oracle Analytics Publisher integration are described here for general information, but these aren't recommended for general use and manipulation of these methods could cause the reporting integration to fail.

For methods where a delivery channel needs to be defined on the publisher server, see the *Administering Oracle Analytics Publisher in Oracle Analytics Server* guide.

XMLP Driver Service Methods

Method	Description	Exposed
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 Oracle Publisher server. It calls the RunBIPReport method to run the report.	Yes
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 Oracle Publisher server. It calls the RunBIPReport method to run the report.	Yes
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 Oracle Publisher server to deliver the email message.	Yes
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 Oracle Publisher server to send the FTP.	Yes
FaxReport	Submits a scheduled report, and then faxes this report when the report finishes. You must configure Oracle Publisher server so that it sends this fax to the fax computer.	Yes
LocalDeliveryReport	Saves a report in Oracle Publisher repository that resides on Oracle Publisher file server.	Yes
PrintReport	Submits a scheduled report, and then prints this report when the report finishes. You must make sure you configure Oracle Publisher server to use this printer.	Yes
ScheduleReport	Schedules a report.	Yes
WEBDAVReport	Submits a scheduled report, and then sends this report to the domain according to the WEBDAV that you configure on Oracle Publisher server.	Yes
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.	No

Method	Description	Exposed
	Attach the report to an Attachment business component. The AttachEntity method is part of the GenerateBIPReport method.	

Create a Business Object

To generate a report using a workflow, you need to create a new record in the Report Output business component (BC). To create a new record in the Report Output BC, you must add a new business object.

Here are the steps:

1. Create a new business object with these values:
 - o Name: Report Output
 - o Primary Business Component: Report Output BC
 - o Query List Business Component: Query List
2. Add these business object components:
 - o Report Output BC
 - o Report Output BC.Sequence Number (Sequence)

Create a Workflow Process

In Siebel Tools, create a new workflow processes for Siebel BI Publisher report generation.

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

BIP Create Report Output Workflow Process

Create the new workflow process, BIP Create Report Output, with these properties:

BIP Create Report Output Workflow Process Details

Workflow Process Name	Workflow Mode	Business Object
BIP Create Report Output	Service Flow	Report Output

In this table, the attributes are these same for these values:

- Business Object: Report Output
- Data Type: String

BIP Create Report Output Workflow Process Properties

Name	Description	In/Out
Report Name	New report name to be created.	In
ReportOutputId	Row ID of Report Output BC for the generated report.	Out

BIP Create Report Output Workflow Process Properties

Name	Type	Business Component	Operation
Start	Start		
New Report	Siebel Operation	Report Output BC	Insert
End 0	End		

BIP Create Report Output Workflow Process Step Branch

Step Name	Name	Type	To Step Name
Start	Connector 0	Default	New Report
New Report	Connector 1	Connector	End 0

BIP Create Report Output Workflow Process Step I/O Arguments

- Name: Report Name
Type: Process Property
Field Name: Report Name
Sub Process Input Argument: Report Name
Method Argument: Report Name
Property Name: Report Name
Input Flag: I
- Name: ReportOutputFileName
Type: Process Property
Field Name: ReportOutputFileName
Sub Process Input Argument: ReportOutputFileName
Method Argument: ReportOutputFileName
Property Name: Report Name
Input Flag: I
- Name: Owner Id
Type: Expression
Field Name: Owner Id
Sub Process Input Argument: ReportOutputFileName
Method Argument: ReportOutputFileName
Value/Search Specification: LoginId()
Input Flag: I
- Name: Time Generated
Type: Expression
Field Name: Time Generated
Sub Process Input Argument: Time Generated
Method Argument: Time Generated
Value/Search Specification: TimeStamp()

Input Flag: I

- Name: O:New Report70

Type: Expression

Field Name: O:New Report70

Sub Process Input Argument: O:New Report70

Method Argument: O:New Report70

Property Name: ReportOutputId

Value/Search Specification: [&Siebel Operation Object Id]

Input Flag: O

BIP Report Generation Workflow Process

Create the new workflow process, BIP Report Generation, with these properties:

BIP Report Generation Workflow Process Details

Workflow Process Name	Workflow Mode
BIP Report Generation	Long Running Flow

In this table, the attributes are these same for these values:

- Data Type: String

BIP Report Generation Workflow Process Properties

Name	Description	In/Out
BIP Password	BI Publisher server password for session authentication.	In
BIP User ID	BI Publisher server User ID for session authentication.	In
Bookmark	The user-specified search specification on the active view. For example: [Name] Like '3Com*'	In
Integration Object	Name of the Integration Object for the primary business object.	In
Language Code	Language code where the Siebel Object Manager is running.	In
Report Name	Language-independent report name to be generated.	In

Name	Description	In/Out
Report Output Type	The output format the report must be generated in.	In
ReportOutputId	Row Id of Report Output BC for the generating report.	In/Out
ViewMode	Visibility of the current active view. For example: Organization.	In
reportFileName	Temporary file generated for the Siebel data in the BI Publisher Server.	Out

BIP Report Generation Workflow Step

- Name: Start
Type: Start
Business Component: Report Output BC
- Name: BIP New Report Output
Type: Sub Process
Sub Process Name: BIP Create Report Output
- Name: RunReport
Type: Business Service
Business Service Name: XMLP Driver Service
Business Service Method: GenerateBIPReport
- Name: End 0
Type: End

BIP Report Generation Workflow Step Branch

- Step Name: Start
Name: Connector 0
Type: Default
To Step Name: BIP New Report Output
- Step Name: BIP New Report Output
Name: Connector 2
Type: Connector
To Step Name: Run Report
- Step Name: Run Report
Name: Connector 1
Type: Connector
To Step Name: End 0

BIP Report Generation Workflow Step I/O Arguments

- Step Name: BIP New Report Output
Name: Report Name
Type: Process Property
Field Name: Report Name
Sub Process Input Argument: Report Name
Method Argument: Report Name
Property Name: Report Name
Input Flag: I
- Step Name: BIP New Report Output
Name: O:BIP New Report Output20
Type: Output Argument
Field Name: O:BIP New Report Output20
Sub Process Input Argument: O:BIP New Report Output20
Method Argument: O:BIP New Report Output20
Property Name: ReportOutputId
Input Flag: O
Output Argument: ReportOutputId
SubProcess Output Argument: ReportOutputId
- Step Name: RunReport
Name: LoginId
Type: Expression
Field Name: LoginId
Sub Process Input Argument: LoginId
Method Argument: LoginId
Value/Search Specification: LoginId()
Input Flag: I
- Step Name: RunReport

Name: BIPUserId

Type: Process Property

Field Name: BIPUserId

Sub Process Input Argument: BIPUserId

Method Argument: BIPUserId

Property Name: BIPUserId

Input Flag: I

- Step Name: RunReport

Name: BIPPassword

Type: Process Property

Field Name: BIPPassword

Sub Process Input Argument: BIPPassword

Method Argument: BIPPassword

Property Name: BIPPassword

Input Flag: I

- Step Name: RunReport

Name: ReportName

Type: Process Property

Field Name: ReportName

Sub Process Input Argument: ReportName

Method Argument: ReportName

Property Name: ReportName

Input Flag: I

- Step Name: RunReport

Name: ReportOutputType

Type: Process Property

Field Name: ReportOutputType

Sub Process Input Argument: ReportOutputType

Method Argument: ReportOutputType

Property Name: ReportOutputType

Input Flag: I

- Step Name: RunReport

Name: LanguageCode

Type: Process Property

Field Name: LanguageCode

Sub Process Input Argument: LanguageCode

Method Argument: LanguageCode

Property Name: LanguageCode

Input Flag: I

- Step Name: RunReport

Name: ReportOutputId

Type: Process Property

Field Name: ReportOutputId

Sub Process Input Argument: ReportOutputId

Method Argument: ReportOutputId

Property Name: ReportOutputId

Input Flag: I

- Step Name: RunReport

Name: ReportSelected

Type: Process Property

Field Name: ReportSelected

Sub Process Input Argument: ReportSelected

Method Argument: ReportSelected

Property Name: Report Name

Input Flag: I

- Step Name: RunReport

Name: IntegrationObject

Type: Process Property

Field Name: IntegrationObject

Sub Process Input Argument: IntegrationObject

Method Argument: IntegrationObject

Property Name: IntegrationObject

Input Flag: I

- Step Name: RunReport

Name: ViewMode

Type: Process Property

Field Name: ViewMode

Sub Process Input Argument: ViewMode

Method Argument: ViewMode

Property Name: ViewMode

Input Flag: I

- Step Name: RunReport

Name: MethodName

Type: Literal

Field Name: MethodName

Sub Process Input Argument: MethodName

Method Argument: MethodName

Value/Search Specification: GenerateBIPReport

Input Flag: I

- Step Name: RunReport

Name: Bookmark

Type: Process Property

Field Name: Bookmark

Sub Process Input Argument: Bookmark

Method Argument: Bookmark

Property Name: Bookmark

Input Flag: I

- Step Name: RunReport

Name: LDAPEnabled

Type: Literal

Field Name: LDAPEnabled

Sub Process Input Argument: LDAPEnabled

Method Argument: LDAPEnabled

Value/Search Specification: True

Input Flag: I

Deploy and Test the Configuration

As a final task, deploy and test the configuration.

1. Compile the SRF.
2. Sign back into the application.
3. Activate the new workflow processes in the application. Go to Site Map > Administration - Business Process > Workflow Deployment > Active Workflow Processes.
4. Query for **BIP***.
5. Select and Activate.
6. To test, follow the test steps detailed at the beginning of the document: *Test the Siebel BI Publisher 8 Reports Business Service Setup*.

Activate Siebel BI Publisher Reports Business Service (8.1.1.7+)

The Siebel BI Publisher Reports business service (8.1.1.7+) lets you run Oracle BI Publisher reports from Siebel using a workflow process or through scripting. Note that BI Publisher is now rebranded as Oracle Analytics Publisher.

Reports are generated in the background by connecting to the BI Publisher/Analytics Publisher server. The report output is stored in the Siebel file system and accessed from the My BI Publisher Reports view. Alternatively, using appropriate methods, the report can be attached to an entity or sent to a particular delivery channel.

Note: The required objects for the Reports business service are automatically included with the Siebel Fix Pack installation. However, an SRF compilation is required to deploy the new business object and business service and to test the workflow.

Siebel BI Publisher/Analytics Publisher Reports Business Service Methods

Several Siebel BI Publisher Reports business service methods are available with the Reports business service in the 8.1.1.7 Fix Pack and above. See *Reports Business Service Methods You Can Use with the XMLP Driver Service* for details.

Note: Methods used **internally** in Siebel's Oracle Analytics Publisher integration are described here for general information, but these aren't recommended for general use and manipulation of these methods could cause the reporting integration to fail.

For methods where a delivery channel needs to be defined on the publisher server, see the *Administering Oracle Analytics Publisher in Oracle Analytics Server* guide.

8 Customizing Siebel CRM Reports

Customizing Siebel Reports

This chapter describes how to customize Siebel reports. It includes these topics:

- *Process of Creating Custom Reports*
- *Specifying the Siebel Data That Report Templates Use*
- *Defining Optional Fields When Registering Report Templates*
- *Customizing Reports That Use Parameters*

Process of Creating Custom Reports

The process to create a custom report involves these tasks:

Tasks Required to Create Custom Reports

These tasks are required when you create custom reports:

1. Create XML data in the Sample Data File Generation view in the Siebel application.
2. Create a new or modify an existing report template. For more information, see *Registering Report Templates*.
3. Register layout templates. For more information, see *Associating Report Templates with Views*.
4. Associate the report template with Siebel views. For more information, see *Associating Report Templates with Views*.
5. Change the report sequence order in the View Association view in the Siebel application.

Optional Tasks to Create Custom Reports

You can also do these optional tasks to meet your business needs:

- Create a new integration object. You do this if an appropriate integration object doesn't exist, or to extend existing integration objects in Siebel Tools. For more information, see *Creating Report Templates*.
- Add multiple integration objects to a report.
- Share report layout templates. For more information, see *Sharing Report Templates*.
- Define report layout templates for selected records.

Creating Report Templates

This topic describes how to use Oracle Publisher Desktop to create a report template.

For more information, see *Oracle Publisher Desktop*.

This task is a step in *Process of Creating Custom Reports*.

Prerequisites

Determine if you can use a predefined report template.

It's recommended that you create a new report template only if no predefined report template meets your report layout requirements. Here 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 need multiple versions of the same report where each version displays slightly different data to different users.

Here 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](#).

Create a Report Template

1. Install Oracle Publisher Desktop.

Note: As a best practice, use the same version of Oracle Publisher Desktop and Oracle Publisher Server. For example, if you're using Oracle BI Publisher Desktop 11/Oracle Analytics Publisher Desktop 5.5, then you should run it with Oracle BI Publisher Server 11/Oracle Analytics Server 5.5.

2. Start Microsoft Word.
3. Click the Publisher menu, click Data, and then select the Load Sample XML Data menu item.
4. In the Select XML Data window, select an XML data file.

XML data files reside in this folder on the Siebel Server: `SIEBEL_SERVER_ROOT\XMLP\DATA`.

For more information, see [Directory Structure That Siebel Reports Uses](#).

5. Use the Template Wizard to define the report template:
 - a. In the Microsoft Word toolbar, select Insert, Table/Form, and then Wizard.
 - b. In the Template Wizard, select one of these report formats, and then click Next:
 - Table
 - Form
 - Free Form
 - c. (Optional) Change the XML data group, and then click Next.
 - d. Select 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 these 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.

- [illegible]

The report template is saved as an XLIFF file.

- ## 6. Preview the report template:

- a.** In Microsoft Word, open the template file.

Siebel CRM stores the template files in this folder: `SIEBSRVR_ROOT\XMLP\TEMPLATES`.

- b.** Click the Publisher menu, click Preview Template, and then select the output file format.

Siebel CRM displays the report in the format that you select.

- c. Preview different output file formats.

- 7.** Copy the report template files to the Siebel Server:

- If you're using a disconnected client, then copy these files from the 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're using a connected client, then save the template file in RTF format, and then copy this RTF file to the Siebel Server in this folder: `SIEBSRVR_ROOT\XMLP\TEMPLATES`

Make sure you don't include any special characters in the report name. For more information about these folders, see *Directory Structure That Siebel Reports Uses*.

Registering Report Templates

You must register report templates so that users can select them in the Run Report pane.

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

This task is a step in *Process of Creating Custom Reports*.

Register a Report Template

1. Make sure Oracle Publisher Server is running:
 - a. On the machine where Oracle Publisher Server is installed, select Start, click Programs, and then select, for example, the `Oracle - BIPHomeX` menu item.
 - b. Select Start Publisher.

A command window opens, which you must leave open while the Oracle Publisher Server is running.

2. Sign 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](#).

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 select an integration object in the child Integration Objects list. For more information, see *Specifying the Siebel Data That Report Templates Use*.
6. Select the report template file that Siebel CRM uses to run the report:
 - a. In the Template field, select the Magnifying Glass.
 - b. In the Add Attachment dialog box, click Choose File and then navigate to the following folder:
`SIEBSRV_ROOT\XMLP\TEMPLATES`
Siebel CRM stores report template files in this folder. For more information, see *Directory Structure That Siebel Reports Uses*.
 - c. Choose the report template file that this report must use, and then click Add.
For example, choose the file you saved in *Creating Report Templates*.
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*.
8. Upload the files:
 - o If you use a connected client, then click Upload Files.
 - o If you use a disconnected client, then locate and upload the RTF and XLIFF files.
Siebel automatically generates the XSL file and uploads these three files to the Siebel Server and Publisher XDO Engine.For more information about how to upload files, see *How Siebel Uploads Report Templates*.
9. Verify that Siebel CRM uploaded the report:
 - a. Sign in to Oracle 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 Controls Access to Reports*.

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 Uploads Report Templates

The typical steps involved when Siebel CRM uploads the files associated with a report template when you register this template are as follows:

1. You use Oracle 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:
 - Uses the PublicReportServiceService Web service to do the upload.
 - Creates a new folder structure in Oracle 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.

Oracle 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 Runs Reports in Connected Clients* and *Oracle Publisher Desktop*.

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

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

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

All RTF, XLIFF, and XSL files are local to the Publisher XDO Engine when Siebel CRM runs the report. For more information, see *How Siebel Runs Reports in Disconnected Clients*.

-
2. In a connected client, Siebel CRM uses the PublicReportServiceService Web service to store the RTF and XLIFF files in Oracle 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`

Siebel CRM Validation Checks When You Upload a Report Template

Siebel CRM carries out the following validation checks 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 Displays Reports According to the View*.

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

This task is a step in *Process of Creating Custom Reports*.

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*.

6. Test the new report.

For more information, see *Running Reports*.

Specifying the Siebel Data That Report Templates Use

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

- *Modifying Integration Objects*
- *Creating XML Files from Integration Objects*
- *Reducing the Amount of Data That Integration Objects Transfer*

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*.

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*.

3. In the Integration Objects list, click New.
4. Specify the integration object, using values from the following table.

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

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*.

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

The following table 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*.

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

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

The following items apply for each row in the previous table.

- The Position automatically preserves visibility, organization visibility, and view mode. For information, see *How Siebel Controls Access to Reports*.
- 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*.
- 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*.
- 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*.

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*.

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*.

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*.

- 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 into the External Name, External Data Type, External Length properties.
 - Set the XML Sequence property.
 - Set the XML Tag property. You typically set the XML Tag property to the same value that the Name property contains.

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

ss_

3. Reduce the Size of the integration object that you modified in Step 2.

For more information, see *Reducing the Amount of Data That Integration Objects Transfer*.

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

Creating New Integration Objects to Add Fields to Reports

Siebel CRM populates a QueryString field in the sample XML and parses it as a field value in the integration object while the report runs, by default. This field contains the user-defined query that the user entered to run the report. You

can display this field value on the report template. For example, the format of the field might be [Account Status] = "Active"].

To create a new integration object to add fields to reports

1. Make sure no predefined integration objects exist that meet your report layout requirements.
Siebel CRM comes with predefined integration objects. It is recommended that you create a new integration object only if the predefined integration objects do not meet your requirements. For more information, see *Modifying Predefined Integration Objects to Add Fields to Reports*.
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*.
 - 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*.
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*.
7. Deliver your changes.
8. Use the Siebel Migration Application Incremental Runtime Repository Data Service to migrate integration object changes to your target environment.
For more information about Siebel Migration and the Migration Application Incremental Runtime Repository Data Service, see *Siebel Database Upgrade Guide*.

Creating XML Files from Integration Objects

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

To create XML files from integration objects

1. Log in to the Siebel client with administrator privileges.
2. Navigate to the Administration - BIP Publisher Reports screen, and then the Sample Data File Generation view.
3. In the Sample Data File Generation list, choose an integration object.
4. Click Generate Sample XML.
Siebel CRM creates an XML file and names it using the following format: `integration_object_name.xml`. For example:
`BIPLiteratureFulfillment.xml`
5. Save the file in the following folder:
`SIEBSRV_ROOT\XMLP\DATA`
You can now create a report template that uses this XML data. For more information, see *Creating Report Templates*.

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 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 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 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 *Oracle Publisher Desktop*. For more information about creating an XML schema, see *Transports and Interfaces: Siebel Enterprise Application Integration*.

Reducing the Amount of Data That Integration Objects Transfer

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

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

To reduce the amount of data that integration objects transfer

- Create a separate integration object for each report.

Do not create a single, large integration object that multiple reports use.
- Deactivate every integration component field that the report does not require.

If you use the EAI Siebel Wizard to create an integration object, then it adds all business component fields to the integration component. You must deactivate fields that the report does not require after the wizard finishes running.

- Do not flag all business component fields as Force Active. When a business field is flagged as Force Active, this loads the business component field during each query. If a minimal number of fields are flagged as Force Active, this improves performance.

Defining Optional Fields When Registering Report Templates

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

- *Allowing Users to Choose The Records That a Report Contains*
- *Sharing Report Templates*

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*.

3. Configure the default language for the report template:

- o 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* and *Creating Multilingual Reports*.
- o Select 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:

```
SIEBSEVR_ROOT\XMLP\TEMPLATES
```

Siebel CRM stores XLIFF files in this folder. For more information, see *Directory Structure That Siebel Reports Uses*. 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*.

- 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*.

5. Add report parameters.

For more information, see *Customizing Reports That Use Parameters*.

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 find problems using this feature, then you might need to do some configuration in Siebel Tools.

To allow users to choose the records that a report contains

1. Sign in to the Siebel client with administrator privileges.
2. Navigate to the Administration - BIP Publisher Reports screen.
3. Find the report template you must change.

For more information, see *Views You Use to Register Report Templates*.

4. In the Selected Records field, do one of the following:
 - o **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*.
 - o **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 Controls Access to Reports*.

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](#).

- 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](#)
- [Adding Report Parameters to Report Templates](#)
- [Adding Report Parameters to Oracle Publisher Desktop](#)
- [Configuring Report Parameters to Filter Data](#)
- [Report Parameter Types You Can Specify](#)
- [Adding Scrollbars to the Parameters Section of the Run Report Pane](#)
- [Modifying the Sequence of Report Names in the Run Report Pane](#)

About Report Parameters

Note: A *report parameter* is a type of filter that refines the data Siebel CRM sends to a report template file that Oracle 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](#).

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

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 to Oracle Publisher Desktop.
For more information, see [Adding Report Parameters to Oracle Publisher Desktop](#).
2. Log in to the Siebel client with administrator privileges.
3. Navigate to the Administration - BIP Publisher Reports screen, then the Reports - Custom Templates view.
For more information, see [Views You Use to Register Report Templates](#).
4. In the Templates list, choose a template, and then add a check mark to the Parameters field.
This field makes sure Siebel CRM displays the Parameters section in the Run Reports pane.
5. In the Parameters list, click New.
To avoid a problem, it is recommended that you use the New button instead of Copy Record.
6. Specify values for the report parameter, using information from the following table.

Field	Description
Label	Enter text that describes the report parameter. Siebel CRM displays the text you enter in the Parameter section of the Run Report pane.
Name	Enter a name for the report parameter. The value you enter field must reference a parameter definition that exists in the report template. For example, if you set the Name field to Param1, then the report template must include a definition for Param1.
Order	Enter a number that indicates the sequence that Siebel CRM uses to display report parameters in the Parameter section of the Run Report pane. Siebel CRM sequences these report parameters in ascending numeric order. For example, if the Parameters list includes Param1 and Param2, and if you set the Order for Param1 to 1 and Param2 to 2, then Siebel CRM displays Param1 immediately before Param2. If the Order field is empty, then Siebel CRM displays this report parameter after the report parameters that include a value in the Order field.
Type	Specify the type of parameter. For more information, see Report Parameter Types You Can Specify .

Field	Description
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 you want Siebel CRM to 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 7a through Step 7c 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:

- o 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.
 - o 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](#).

Adding Report Parameters to Oracle Publisher Desktop

The following procedure describes how to add report parameters to Oracle Publisher Desktop.

To add report parameters to Oracle Publisher Desktop

1. In Oracle Publisher Desktop, 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*.

3. Make sure that a definition exists in the Siebel application for the parameter you added in Step 6.
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 Publisher Desktop, such as Param1. For more information about parameter syntax and usage, see *Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>) or *Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server* available at <https://docs.oracle.com/en/middleware/bi/analytics-server/design-publish/index.html>.
4. (Optional) Configure Siebel CRM to display the parameter settings that the user provides.

To verify that Oracle 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 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*.

Report Parameter Types You Can Specify

The following table describes the report parameter types that you can specify in the Type field of the Parameters list.

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.
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 edge (left) of the Run Report pane. It uses the longest label that you define as the starting position. It aligns all other parameter types relative to this starting position. 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.

Guidelines for Defining Parameters in Report Templates

If you define report parameters in a report template, then adhere to the following guidelines:

- Siebel CRM sends each report parameter value that a user enters to Oracle 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

Data Type	Number of Parameter Fields
Number	10
Date	5
Date/Time	5
Check box	10

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 Displays Reports According to the View*.

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.

9 Customizing Master-Detail Reports

Customizing Master-Detail Reports

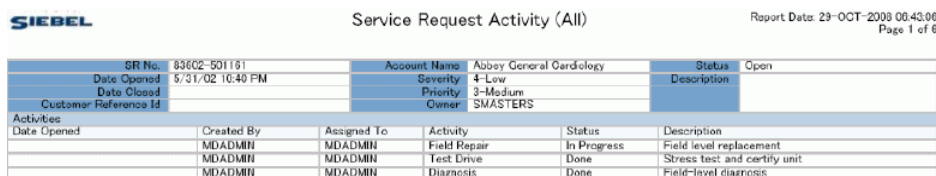
This chapter describes how to customize a master-detail report. It includes the following topics:

- *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.

The following figure shows the Service Request Activity - All report in Siebel Service, which is an example of a master-detail report.



SR No. 83602-901161		Account Name Abbey General Cardiology		Status Open	
Date Opened	5/31/02 10:40 PM	Severity	4-Low	Description	
Date Closed		Priority	3-Medium		
Customer Reference Id		Owner	SMASTERS		
Activities	Created By	Assigned To	Activity	Status	Description
Date Opened	MDADMIN	MDADMIN	Field Repair	In Progress	Field level replacement
	MDADMIN	MDADMIN	Test Drive	Done	Stress test and certify unit
	MDADMIN	MDADMIN	Diagnosis	Done	Field-level diagnosis

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*.

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*.

Many reports that come predefined with Siebel CRM are master-detail reports.

How Siebel Handles Multivalue Fields

The business object that the integration object references specifies how the master-detail report incorporates a parent business component and child and grandchild business components. When Siebel CRM runs a report, it captures the user interface context, and then sends it to the primary integration component of the integration object. If a report includes a multivalue field (MVF), then Siebel CRM displays only the first record. To display all the records from a multivalue field, you must create an integration object that references the business component that contains the multivalue field that Siebel CRM must display. For more information, see [How Siebel Reports Uses Integration Objects](#).

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 [About Master-Detail Reports](#).
2. Open Oracle Publisher Desktop.
For more information, see [Oracle Publisher Desktop](#).
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](#)
This template, for example, includes the following items:

- o Table for the master record that Siebel CRM displays in a form.
- o Table for each set of child records that Siebel CRM displays in a list.
- o For-each loop that Siebel CRM runs for all master records. It includes the parent form and all child lists.
- o 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](#).

4. In the Publisher menu, select Tools, select Field Browser, and then select Show All.

The Field Browser dialog box displays, that allows you to view and modify the template. For more information, see [Oracle Publisher Code for Service Request Activity \(All\) Report Template](#).

Oracle Publisher Code for Service Request Activity (All) Report Template

The following table describes the code that Oracle Publisher uses for the Service Request Activity (All) report template. You can view this code if you use Oracle Publisher Desktop to open the Service Request Activity (All) report template. For more information about this code, see the *Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher* available on Oracle Technology Network (<http://www.oracle.com/technetwork/indexes/documentation/index.html>) or *Designing and Publishing Pixel-Perfect Reports in Oracle Analytics Server* available at <https://docs.oracle.com/en/middleware/bi/analytics-server/design-publish/index.html>. For more information about integration objects, see [How Siebel Reports Uses Integration Objects](#).

Syntax	Oracle Publisher Code	Description
If	<code><?if://ssServiceRequest?></code>	An IF condition that determines if service request record exist, do not exist, or are not displayed.
for-eachgroup: ssServiceRequest	<code><?for-each-group:ssServiceRequest;position()??></code>	The start of a group section that iterates for each record in the Service Request data set.
ssSrNumber ssAccount ssStatus	<code><?ssSrNumber?> <?ssAccount?> <?ssStatus?</code>	Field mappings.
If	<code><?if://ssCreated?></code>	If condition that does not display the ssCreated field if null.
ssCreated	<code><?format-date:psfn:totext(ssCreated,"yyyy-MM-dd'T'HH:mm:ss","MM/dd/yyyy hh:mm:ss");'SHORT_TIME'??></code>	Field formatting.
End	<code><?end if?></code>	End of if condition.

Syntax	Oracle Publisher Code	Description
ssSeverity ssDescription	<code><?ssSeverity?> <?ssDescription?></code>	Field mappings.
If	<code><?if://ssClosedDate?></code>	If condition that does not display the ssClosedDate field if null.
ssClosedDate	<code><?format-date:psfn:totext(ssClosedDate,"yyyy-MM-dd'T'HH:mm:ss","MM/dd/yyyy hh:mm:ss");'SHORT_TIME'?></code>	Field formatting.
End	<code><?end if?></code>	End of if condition.
ssPriority ssCustomerRefNumber ssOwner	<code><?ssPriority?> <?ssCustomerRefNumber?> <?ssOwner?></code>	Field mappings.
for-each:current-group()	<code><?for-each:current-group()?></code>	Start of a group section, iterating for each record in the current group (for example, Service Request).
If	<code><?if:current-group()//ssAction?></code>	An IF condition that determines if action records exist, do not exist, or are not displayed.
for-each:ssAction	<code><?for-each:ssAction?></code>	Beginning of the repeating element group for Action.
If	<code><?if://ssCreated?></code>	If condition that does not display the ssCreated field if null.
ssCreated	<code><?format-date:psfn:totext(ssCreated,"yyyy-MM-dd'T'HH:mm:ss","MM/dd/yyyy hh:mm:ss");'SHORT_TIME'?></code>	Field formatting.
End	<code><?end if?></code>	End of if condition.
ssCreatedByN ssOwnedBy sssStatusType ssDescription	<code><?ssCreatedByName?> <?ssOwnedBy?> <?ssType?> <?ssStatus?> <?ssDescription?></code>	Field mappings.
end for-each:ssAction	<code><?end for-each?></code>	End of the repeating element group for Action.
End if	<code><?end if?></code>	The end of the if condition
end for-each	<code><?end for-each?></code>	End of current-group() repeating group section.
page break	<code><?split-by-page-break:?></code>	Page break after the end of the group section.

Syntax	Oracle Publisher Code	Description
end for-each-group: ssServiceRequest	<code><?end for-each-group?></code>	End of Service Request group section.
End	<code><?end if?></code>	End statement that ends a subroutine.
If	<code><?if:not(// ssServiceRequest)?></code>	If condition that prints "No Data Found" in the report if there are no records.
End	<code><?end if?></code>	End if statement for the condition in the first row.

Creating a Master-Detail Report Template in Microsoft Word

Creating a master-detail report follows the same process as creating a simple report. However, consider the following important points when creating master-detail reports:

- Make sure that all child integration components in the integration object include the required value in the Parent Integration Component property. For more information, see *How Siebel Reports Uses Integration Objects*.
- 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*.
- 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 Publisher Desktop in Microsoft Word, and then create a new document.
For more information, see *Oracle Publisher Desktop*.
2. In the Publisher menu, click Data and then click 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 Publisher 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 Publisher menu, click Insert and then click Repeating Group.
- c. Choose the defaults, and then click OK.

7. Insert a page break after the final detail section:

- a. Select the Oracle Publisher properties for 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](#).

9. Register the report.

For more information, see [Registering Report Templates](#).

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 Publisher syntax.

To view the descriptive text for the repeating group sections

1. Open Oracle Publisher Desktop in Microsoft Word.
2. In the Publisher menu, click Options, and then Build.
3. In the For-each form field box, choose Descriptive.

10 Deploying Reports

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*
- *Deploying Integration Objects to the Production Environment*

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*
2. *Packaging Report Files in the Development Environment*
3. *Deploying Report Files to the Production Environment*
4. *Packaging Report Database Records in the Development Environment*
5. *Deploying Report Database Records to the Production Environment*
6. (Conditional) *Deploying Integration Objects to the Production Environment*

Preparing the Deployment Environment

This task is a step in *Process of Deploying Reports*.

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*.
 - 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*.

Packaging Report Files in the Development Environment

The following procedure describes how to package report files in the development environment.

This task is a step in *Process of Deploying Reports*.

To package 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.

- i. Copy the XML files to the following folder:

```
SharedFolder\PackageName\file\AppServer\XMLP\DATA
```

- ii. Copy the RTF and XSL files to the following folder:

```
SharedFolder\PackageName\file\AppServer\XMLP\TEMPLATES
```

- iii. 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:

```
admpkg 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 Report Files to the Production Environment

The following procedure describes how to deploy the report files to the production environment.

This task is a step in *Process of Deploying Reports*.

To deploy 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 Step 2 of the previous topic.
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 Step 1 in the previous topic are available in the correct folders.

Packaging Report Database Records in the Development Environment

The following procedure describes how to use Application Deployment Manager (ADM) to package report database records.

This task is a step in *Process of Deploying Reports*.

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:

- o Named *Test Account List*
- o References the BIP Accounts - Current Query integration object
- o 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

The following procedure 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 in the previous topic, 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](#).

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

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

11 Troubleshooting Siebel Reports

Troubleshooting Siebel Reports

This chapter describes how to troubleshoot Siebel Reports. It includes the following topics:

- *Enabling and Disabling Logging for Siebel Reports*
- *Fixing Class Not Found Errors*
- *Troubleshooting Error Messages for Siebel Reports*

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*
- *Enabling Logging and Debugging on Oracle Publisher Server*
- *Enabling and Disabling Debugging in Disconnected Clients*

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) - as described in the following procedures. You can adjust the log levels at any time.

Oracle Publisher logs all exceptions and debug level information.

The following procedure shows how to use the Server Manager UI to set log levels for the XMLP Report server component.

To set log levels for XMLP Report Server component using the Server Manager UI

1. Log in to the Siebel client with administrator privileges.
2. Navigate to the Administration - Server Configuration screen, Servers view, then the Components view.
3. In the Component field, query for XMLP Report Server, and then click the Events tab.
4. In Events list, query for XMLP Report Log, and then set the value for the log level to 5.

Note: Setting log levels to a high value for all components in report generation – such as, the requesting Object Manager, XMLP Report Server component, EAI Object Manager, File System Manager and Oracle Publisher Server – affects performance. The higher the log level value, the slower the performance.

5. Click the Component tab.
6. In the Component field, query for the Application Object Manager, and then click the Events tab.

For example, query for the following Application Object Manager:

Call Center Object Manager (ENU)

7. In the Events list, query for XMLP Report Log, and then set the value for the log level to 5.
8. Stop, and then restart the XMLP Report Server and Siebel Application Object Manager server components.

For more information about restarting the Siebel Server, see *Siebel System Administration Guide*.

The following procedure shows how to use the command-line interface to set log levels for the XMLP Report server component.

To set log levels for XMLP Report Server component using the command-line interface

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:

- o **gateway computer name** is the name of the computer where the Siebel Gateway.
- o **port number** identifies the port number where the Siebel Gateway is listening. The colon and the port number are optional if using an optional default port.
- o **xmlp siebel server name** is the name of the server where the XMLP Report server component is enabled.
- o **user name** is the login name of the administrator.
- o **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 **s** 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 Oracle Publisher Server

This topic describes how to enable logging and debugging on Oracle Publisher Server.

Enabling Logging on Oracle Publisher Server

You can configure Oracle 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.

Enabling Debugging on Oracle Publisher Server

You can configure Oracle Publisher Server to capture detailed information about errors that occur while working with Siebel Reports in the Siebel Web Client. This debugging is in addition to the typical debugging capabilities that are available in Siebel CRM.

For more information about enabling debugging on Oracle Publisher Server, see the Oracle Analytics Publisher (formerly Oracle BI Publisher) documentation.

Enabling and Disabling Debugging for Disconnected Clients

You can configure the 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 Oracle Publisher report template files.

Use the following procedure to enable debugging for the Publisher XDO Engine.

To enable debugging for 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.

Use the following procedure to disable debugging for the Publisher XDO Engine. Typically, you disable debugging for the Publisher XDO Engine to optimize Siebel Reports performance in a disconnected client.

To disable debugging for 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 *class not found* errors when you preview a report in Microsoft Word. The guidelines for fixing *class not found* errors, when previewing reports in Microsoft Word, are as follows:

- 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
```

The following procedure shows 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 the error, you create a custom batch file that loads the required template libraries before Microsoft Word opens the report template file. You replace the explicit drive and folder locations in this batch file with the drive and locations that your computer uses according to where you installed the Siebel Web Client. You are not required to associate a .doc extension with this batch file.

To fix class not found errors when previewing reports in Microsoft Word

1. Navigate to the following directory on the computer where you installed the Siebel client:

```
C:\Program Files
```

2. Create a new file named `MSWordForBIPub.bat`.
3. Add the following code to the file you created in Step 2:

```
echo %1
```

```
set _JAVA_OPTIONS=-Xbootclasspath/a:
```

```
C:\81DQSSIA\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 necessary, make sure that 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 an RTF file, select Open With, and then select a text editor.
- d. Click Browse to find the .bat file (on your local machine), 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 Publisher requires to publish a report in Microsoft Word.

If you double-click the RTF file, then a command prompt window opens. You can also use the desktop shortcut and run the shortcut to the .bat file to test the report preview.

Troubleshooting Error Messages for Siebel Reports

The topic provides guidelines for fixing some common Siebel Reports error messages.

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 Oracle 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).
Siebel 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).
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 aren't correct. It might occur in a connected or disconnected client. To fix this problem, see Fixing Incorrect Configuration File Setting Errors (SBL-DAT-00322).
SBL-RPT-50504: Error occurred while saving the record. No	This error might occur if you don't provide a sample XML data file for creating an XLIFF file. It occurs only in a disconnected client.

Symptom or Error Message	Solution
<code>data file provided for XLIFF generation.</code>	To fix this problem, see Fixing XLIFF File Errors .
Siebel CRM displays error messaged SBL-RPT-50524 or SBL-EAI-50228	To resolve the issue, you need to alter or add this parameter: <code>EAIFileTransportFolders = <folder where the xml file will be generated></code> . Note: Make sure to include ALL affected folders in the parameter. Make sure to set the parameter for ALL affected Siebel Servers.
Siebel CRM displays the following message if you run a report: Class name incorrect.	This error might occur if Oracle Publisher or the JAR files aren't loaded. It occurs only in a disconnected client. To fix this problem, you set the classpath. For more information, see Fixing the CLASSPATH .
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 <code>HTTPSleepTime</code> method argument to fix this problem. For more information, see Modifying the HTTP Sleep Time .
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: invalidusername or password.	This error message appears if you don't provide the correct user ID and password for Oracle Publisher Server in the XMLP Report server component parameters. This problem occurs only in a connected client. To fix this problem, see Fixing Oracle Publisher Server Login Error .
Generic Error in Compression Routine. (SBL-UIF-00227)	Make sure the <code>temp\xmlp</code> folder exists in the <code>SIEBSVR_ROOT</code> folder. If it doesn't, then add it. For more information, see Directory Structure That Siebel Reports Uses . 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 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.

Symptom or Error Message	Solution
SBL-EAI-04308: Operation 'runReport' of Web Service 'http://xmlns.oracle.com/oxp/service/PublicReportService.PublicReportService' at port 'PublicReportService' failed with the following explanation: <pre>"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"</pre>	This error might occur if you run a report that includes missing information on Oracle Publisher Server. A missing .xdo file is an example of missing information. To fix this problem, upload the report template from Oracle Publisher. For more information about uploading report templates, see Registering Report Templates. This problem occurs only in a connected client.
A 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.
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 Oracle 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.
Siebel CRM displays one of the following message when you upload files to Oracle Publisher: SBL-EAI-05010: Class name incorrect or does not extend SiebelBusinessService : <pre>com/siebel/data/SiebelPropertySet -- JVM Exception:java.lang.NoClassDefFoundError com/siebel/data/SiebelPropertySetObjMgrBusServiceI</pre> 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 <code>siebelroot\classes\original</code> folder to the <code>ORACLE_HOME\j2ee\home\applications\xmlpserver\xmlpserve\WEB-INF\lib</code> folder: - 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 Publisher:	This error indicates that Oracle Publisher Server already includes a copy of the file that you're uploading. The uploadReport method doesn't support uploading a new version of an existing file.

Symptom or Error Message	Solution
<code>SBL-EAI-04308: Operation 'uploadReport' of Web Service 'http://xmlns.oracle.com/oxp/service/v11/PublicReportService.PublicReportSe</code> <code>at port 'PublicReportService_v11'</code> <code>failed with the following explanation:</code> <code>"oracle.apps.xdo.webservice.exception.PublicReportService::execute UploadReport Failure: Due to Report with Path [/SiebelCRMReports/Application Activity/Application Activity.xdo] already exist!"</code>	To fix this problem, you must rename or delete the folder for the existing instance of the report on Oracle Publisher Server. Oracle Publisher Server stores the report template files in the following directory: <code>ORACLE_HOME\xmlp\XMLP\Reports\SiebelCRMReports</code> .

Fixing Errors That Occur When Running Siebel Reports (SBL-OMS-00203)

The following procedure shows how to fix errors that occur when 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 more information about these tasks, see *Enabling Logging for Siebel Reports in the Siebel Application* and *Enabling and Disabling Debugging in Disconnected Clients*.

If you create a report template with a database, but then register it with a different database, then the report might fail. The following procedure describes how to fix this problem – by regenerating the XML and running the report.

To regenerate XML and run the report

1. Regenerate the XML data.
2. Reregister the report template.
For more information, see *Registering Report Templates*.
3. Rerun the report.

Fixing Errors That Occur After Clicking Submit in the Run Report Pane (SBL-SRQ-00103)

The following procedure shows how to fix errors that occur 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 Incorrect Configuration File Setting Errors (SBL-DAT-00322)

The following procedure shows how to fix errors that occur due to incorrect configuration file settings.

To fix incorrect configuration file setting errors

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 (:).

The following procedure shows how to fix an error that occurs (due to incorrect configuration file settings) in a disconnected client.

To fix incorrect configuration file setting error in disconnected clients

1. In the application .cfg file, set the following value for the FileSystem parameter:

```
FileSystem \\computer name\fs\att
```

2. Make sure the `fs` folder in the path that you added in Step 1 is shared.
3. In the XMLPReports section of the .cfg file, make sure the following section is defined:

```
[XMLPReports]
XdoDir = /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 Errors SBL-EAI-50228, SBL-DEV-00341, and SBL-EAI-04262

You might occasionally get one of these errors: SBL-EAI-50228, SBL-DEV-00341, or SBL-EAI-04262. To resolve the issue, you need to alter or add the `EAIFileTransportFolders` parameter.

Set the `EAIFileTransportFolders` parameter to the folder where the XML file will be generated.

Note: Make sure to include ALL affected folders in the parameter. Make sure to set the parameter for ALL affected Siebel Servers.

You can set the parameter using the server or in the UI.

Use the Server Manager to Set the `EAIFileTransportFolders` Parameter

1. Change `ent param EAIFileTransportFolders=<\\fileserver\fs1>;<\\fileserver2\fs2>`.
2. Change `param EAIFileTransportFolders=<\\fileserver\fs1>;<\\fileserver2\fs2>` for server `<servername>`.
3. Edit the proper `.cfg` file adding following entries (application `.cfg`, `tools.cfg`):
 - `[EAIFileTransportConfigSubsys]`
 - `EAIFileTransportFolders = <\\fileserver\fs2>;<\\fileserver2\fs2>;C:\temp;C:\users\data`

Note: These directories might include both file share, UNC, and absolute paths.

Use the UI to Set the `EAIFileTransportFolders` Parameter

1. Go to **Siebel Client > Sitemap > Administration - Server Configuration > Enterprises > Parameters**.
2. Search for: `EAIFileTransportFolder`.
3. Add values for: `<\\fileserver\fs1>;<\\fileserver2\fs2>`.
4. Go to **Sitemap > Administration - Server Configuration > Servers > Parameters**.
5. Select the proper server from the Siebel Servers list applet.
6. In the Server Parameters applet, search for: `EAI File Transport Folder List`.
7. Add values for: `<\\fileserver\fs1>;<\\fileserver2\fs2>`.
8. Edit the proper `.cfg` file adding following entries (application `.cfg`, `tools.cfg`):
 - `[EAIFileTransportConfigSubsys]`
 - `EAIFileTransportFolders = <\\fileserver\fs2>;<\\fileserver2\fs2>`

Note: These directories might include both file share, UNC, and absolute paths.

Fixing XLIFF File Errors

The following procedure shows 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. For more information about this view, see *Creating XML Files from Integration Objects*.
3. Navigate to the Administration - BI Reports screen, then the Sample Data File Generation view.
4. Choose the integration object associated with the report, and then click Generate Sample XML.

Fixing the CLASSPATH

The following procedure shows 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*.
2. Run the following command to change the CLASSPATH:

```
Change param CLASSPATH =  
C:\sba81\siebsrvr\CLASSES\Siebel.jar;C:\sba81\siebsrvr\CLASSES\SiebelXMLP.jar;  
C:\sba81\siebsrvr\classes\xdo-core.jar for named subsystem XMLPJvmSubsys
```

Fixing Oracle Publisher Server Login Error

The following procedure shows how to fix Oracle Publisher Server login error.

To fix Oracle Publisher Server login errors

1. Log in to the Siebel client with administrator privileges.
2. Navigate to the Administration - Server Configuration screen, 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 Oracle Publisher Server.
5. Query for the Publisher Server Password, and then enter the correct password for Oracle Publisher Server.

Fixing Alignment Problems in HTML

The following procedure shows how to fix alignment problems that might occur in a report's HTML output.

To fix 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

The following procedure shows how to fix Japanese Font Errors that might occur with Siebel Reports.

To fix Japanese font errors

1. Locate the `msgothic.ttc` file.
2. Copy the file you located in Step 1 to the following folders on the computer where you installed Oracle Publisher Server:
 - `$JAVA_HOME\jre\lib\fonts`
 - `ORACLE_HOME\common\fonts`
3. Log in to Oracle Publisher Server.
4. Navigate to Admin, Runtime configuration, and then Font Mappings.
5. Create the following font mappings:
 - Arial, Normal, msgothic.ttc
 - Arial, Bold, msgothic.ttc

12 Appendix: List of Siebel Reports

Siebel Web Client Reports

Here's a list of Siebel web client reports:

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
Account List	Account List	BIP Accounts - Current Query	aclist	All
Account Profile	CG Account Profile Report	BIP CG Account Profile Report	CS_accntpro	All
Account Promotion Deal Sheet	CG Promotion Plan Details	BIP CG Promotion Plan Details	CS_ProPlanDet	All
Account Service Profile	Account Service Profile	BIP Current Account Service Profile	acsvcpro	All
Actual Visits	Actual Visits	BIP Actual Visits	LS_ACTVIST	All
Admin Product Line List	Admin Product Line List	BIP Admin Product Line - Current Query	adprodli	All
Admin Sales Cycle - Current Query	Admin Sales Cycle - Current Query	BIP Admin Sales Cycle - Current Query	ADSLSCYC	All
Alphabetic Phone List	Contacts - Phone List	BIP Contacts - Alphabetic Phone List	cntphon	All
BIP Expense Report	BIP Expense Report	BIP Expense Report	exprep	All
By Opportunity	Contacts - By Opportunity	BIP Contacts - By Opportunity	cntopp	All
By Sales Rep	Opportunity - Sales Rep	BIP Opportunities - By Sales Rep	opslsrep	All
CUT Trouble Ticket Activity - All	CUT Trouble Ticket Activity - All	BIP CUT Trouble Ticket Activity - All	srvreqaa_tt_cut	All
CUT Trouble Ticket Detail	CUT Trouble Ticket Detail	BIP CUT Trouble Ticket Detail	srvreqdt_tt_cut	All
CUT Trouble Ticket Summary	XMLP CUT Trouble Ticket Summary	BIP CUT Trouble Ticket Summary	srvreqsm_tt_cut	All
Case Overview	PUB GOV Case Overview	BIP PUB GOV Case Overview	PUB_GOV_CASEOVRW	All
Clinical Site Followup Report	Clinical Site Followup Report	BIP Clinical Site Followup Report	LS_FOUPRE	All

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
Clinical Trip Report Without CRF	Clinical Trip Report Without CRF	BIP LS Clinical Trip Report Without CRF	LS_TRNCRF	All
Contact Current Query Personal	Contact current query personal	BIP Contact - Current Query(Personal)	cntlistper	All
Contact List	Contact List	BIP Contacts - Current Query	cntlist	All
Contacts Alphabetic Phone List Personal	Contacts Alphabetic Phone List Personal	BIP Contacts - Alphabetic Phone List(Personal)	cntphonper	All
Current Agreement Detail	Current Agreement Detail	BIP Agreement Detail	agdet	All
Customer Invoice	FS - Customer Invoice	BIP FS - Customer Invoice	custinv	All
Disease Overview	HLS Disease Overview	BIP HLS Disease Overview	HLS_DISEASEOVRW	All
EIM Interface Tables Report	EIM Interface Tables Report	XMLP Repository EIM Interface Tables - Current Query	rposeim	All
ESP Account Plan	Enterprise Selling Process	BIP ESP Account Plan Overview Report	espcov	All
FINS Demo Health Providers by Specialty	FINS Demo Health Providers by Specialty - Healthcare	BIP FINS Demo Health Providers by Specialty - Healthcare	FD_HEALTH_PROVIDERS	All
FINS Investment Banking Client Profile	FINS Investment Banking Client Profile	BIP FINS Investment Banking Client Profile	fincorp_clientib	All
FINS Life Policy - By City	FINS Life Policy - By City	BIP FINS Life Policy - By City	lifepolcity	All
FINS Life Policy - By Sales Rep	FINS Life Policy - By Sales Rep	BIP FINS Life Policy - By Sales Rep	LIFEPOLSLSRP	All
FINS Life Quotes - BY Source	FINS Life Quotes - BY Source	BIP FINS Life Quotes - BY Source	LIFEPOL_BY_SRC	All
FS Below Minimum Inventory Quantity Per Location	FS Below Minimum Inventory Quantity Per Location	BIP FS Below Minimum Inventory Quantity Per Location	belowmin	All
FS Cycle Count Details	FS Cycle Count Details	BIP FS Cycle Count Detail	cyccountdet	All
FS Field Engineer Activity Summary	FS Field Engineer Activity Summary	BIP FS Field Engineer Activity Summary	FEASUM	All
FS Field Engineer Detail	FS Field Engineer Detail	BIP FS Field Engineer Detail	FEADET	All
FS Inventory Cost Detail	FS Inventory Cost Detail	BIP FS - Inventory Cost Detail	incostdet	All
FS Pick Ticket Details	FS Pick Ticket Details	BIP FS Pick Ticket Details	pickticket	All
FS Pick Ticket Details - No Barcode	FS Pick Ticket Details - No Barcode	BIP FS Pick Ticket Details - No Barcode	pickticketnbc	All

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
FS Product List By Location	FS Product List By Location	BIP FS Product List By Location	prodlistloc	All
FS Repair Detail	FS Repair Detail	BIP FS Repair Detail	repdet	All
FS Repair Detail - No Barcode	FS Repair Detail - No Barcode	BIP FS Repair Detail - No Barcode	repdetnbc	All
Group List	HLS Group List	BIP HLS Group List	HLS_GROUP	All
INS All Policies for Contact	INS All Policies for Contact	BIP INS All Policies for Contact	INSALLPOLCON	All
INS Auto Policy Binder	INS Auto Policy Binder	BIP INS Auto Policy Binder	insapbnd	All
INS Auto Policy Change Request	INS Auto Policy Change Request	BIP INS Auto Policy Change Request	insapcrq	All
INS Auto Policy Memorandum	INS Auto Policy Memorandum	BIP INS Auto Policy Memorandum	insapmem	All
INS Auto Policy Receipt of Payment	INS Auto Policy Receipt of Payment	BIP INS Auto Policy Receipt of Payment	insaprop	All
Incident List	PUB HLS Incident List	BIP PUB HLS Incident List	PUB_HLS_INCIDENT	All
Investigative Case Overview	HLS Case Overview	BIP HLS Case Overview	HLS_CASEOVRW	All
Literature Fulfillment	Literature Fulfillment	BIP Literature Fulfillment	LITFUL	All
Loyalty Accrual Dispute Statement	Loyalty Accrual Dispute Statement	BIP LOY Accrual Dispute	accrualdispute	PDF
Loyalty Member Enrolment Statement	Loyalty Member Enrolment Statement	BIP LOY Member Enrolment	memenrolstmt	PDF
Loyalty Member Statement	Loyalty Member Statement	BIP LOY Statement	memstmt	PDF
Loyalty Membership Cancellation Statement	Loyalty Membership Cancellation Statement	BIP LOY Statement	memcancstmt	PDF
Loyalty Membership Renewal Statement	Loyalty Membership Renewal Statement	BIP LOY Statement	memrenewalstmt	PDF
Loyalty Partner Billing Statement	Loyalty Partner Billing Statement	BIP LOY Statement	partnerbillingstmt	PDF
Loyalty Partner Statement	Loyalty Partner Statement	BIP LOY Partner Statement	partnerstmt	PDF
Loyalty Points Adjustment Statement	Loyalty Member Points Adjustment Statement	BIP LOY Points Adjustment	pointsadjustmentstmt	PDF
Loyalty Points Gift Statement	Loyalty Points Gift Statement	BIP LOY Points Purchase Statement	memgiftptsstmt	PDF
Loyalty Points Purchase Statement	Loyalty Points Purchase Statement	BIP LOY Points Purchase Statement	mempurcptsstmt	PDF
Loyalty Points Transfer Statement	Loyalty Points Transfer Statement	BIP LOY Points Purchase Statement	memtrnptsstmt	PDF

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
Loyalty SoM Assessment Statement	Loyalty Member Status Change Statement	BIP LOY SoM Assessment	somassessmentstmt	PDF
Loyalty Tier Change Statement	Loyalty Tier Change Statement	BIP LOY Tier Change	tierchangestmt	PDF
Loyalty Voucher Cancellation Statement	Loyalty Voucher Cancellation Statement	BIP LOY Voucher Statement	vouchercancstmt	PDF
Loyalty Voucher Issue Statement	Loyalty Voucher Issue Statment	BIP LOY Voucher Statement	voucherissuestmt	PDF
MDV	MDV Report	BIP LS Medical MDV Report	LS_MDV	All
Mobile User Summary Report	Mobile User Summary Report	BIP Mobile User Summary Report	MOBUSRSUM	All
Mobile Users Status Report	Mobile Users Status Report	BIP Mobile Users Status Report	MOBUSRSTAT	All
Opportunities - Summary	Opportunities - Summary	BIP Opportunities - Summary	OPSUM	All
Opportunity List	Opportunity List	BIP Opportunity List - Current Query	oplist	All
Order Detail	Order Detail	BIP Order Detail	ordet	All
PPR Membership Application	PPR Membership Application	BIP PPR Membership Application	APPLDET	All
Pharma Account at a Glance	Pharma Account at a Glance	BIP Pharma Account at a Glance	LS_ACATGL	All
Pharma Call Frequency - Account	Pharma Call Frequency - Account	BIP Pharma Call Frequency - Account	LS_FREQAC	All
Pharma Call Frequency - Professional	Pharma Call Frequency - Professional	BIP Pharma Call Frequency - Professional	LS_FREQPF	All
Pharma Contracts	Pharma Contracts	BIP Pharma Contracts	LS_CONTRACT	All
Pharma Formulary List	Pharma Formulary List	BIP Pharma Formulary List	LS_FORMUL	All
Pharma ME Event Agenda	Pharma ME Event Agenda	BIP Pharma ME Event Agenda	LS_MEEVAG	All
Pharma ME Event Invitee	Pharma ME Event Invitee	BIP Pharma ME Event Invitee	LS_MEEVIN	All
Pharma ME Event Professional Attended	Pharma ME Event Professional Attended	BIP Pharma ME Event Professional Attended	LS_MEEVAT	All
Pharma ME Event Sign-Ins	Pharma ME Event Sign-Ins	BIP Pharma ME Event Sign-Ins	LS_ETSIGN	All
Pharma ME Session Roster	Pharma ME Session Roster	BIP Pharma ME Session Roster	LS_MESERT	All
Pharma Managers Call Per Day	Pharma Manager's Call Per Day	BIP Pharma Manager's Call Per Day	LS_MGRPD	All

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
Pharma MedEd Profile	Pharma MedEd Profile	BIP Pharma MedEd Profile	LS_MEEDPF	All
Pharma Meeting Sign-In	Pharma Meeting Sign-In	BIP Pharma Meeting Sign-In	LS_MTSIGN	All
Pharma My Sample History	Pharma My Sample History	BIP Pharma My Sample History	LS_MYSAMP	All
Pharma Objectives	Pharma Objectives	BIP Pharma Objectives	LS_OBJECT	All
Pharma Orders Detail (eBusiness)	Pharma Orders Detail (eBusiness)	BIP Pharma Orders Detail (eBusiness)	Phordet	All
Pharma Orders Summary (eBusiness)	Pharma Orders Summary (eBusiness)	BIP Pharma Orders Summary (eBusiness)	Phorsum	All
Pharma Professional List	Pharma Professional List	BIP Pharma Professional List	LS_PFLIST	All
Pharma Professional at a Glance	Pharma Professional at a Glance	BIP Pharma Professional at a Glance	LS_PROFGL	All
Pharma Protocol Payments	Pharma Protocol Payments	BIP Pharma Protocol Payments	LS_PROTPAY	All
Pharma Sample Adjustment	Pharma Sample Adjustment	BIP Pharma Sample Adjustment	LS_SAMPADJ	All
Pharma Sample History	Pharma Sample History	BIP Pharma Sample History	LS_SAMPHIST	All
Pharma Sample Invoices	Pharma Sample Invoices	BIP Pharma Sample Invoices	LS_INVOICE	All
Pharma Sample Orders	Pharma Sample Orders	BIP Pharma Sample Orders	LS_SAMPORD	All
Pharma Sample Receipt Report	Pharma Sample Receipt Report	BIP Pharma Sample Receipt Report	LS_RECEIPT	All
Pharma Sample Transfer Out	Pharma Sample Transfer Out	BIP Pharma Sample Transfer Out	LS_TRANSOUT	All
Pipeline Report By Rep	Pipeline - By Rep	BIP Pipeline Report By Rep	piperep	All
Planned vs Actual Patient Dates	Planned vs Actual Patient Dates	BIP Planned vs Actual Patient Dates	LS_PLNACT	All
Portfolio Map Report	Portfolio Map Report	BIP Portfolio Map Report	portmap	All
Portfolio Plan Report	Portfolio Plan Report	BIP Portfolio Plan Report	portplan	All
Portfolio Selected Unit Analysis	Portfolio Selected Unit Analysis	BIP Portfolio Selected Unit Analysis	portsu	All
Position List - Current Query	Position List - Current Query	BIP Position List - Current Query	poslstcq	All
Price Lists	Price Lists	BIP Price List - Current Query	pricelst	All
Product Defect Detail	Product Defect Detail	BIP Product Defect Detail	PRODEFDT	All

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Output Type
Protocol Sites Addresses Report	Protocol Sites Addresses Report	BIP Protocol Sites Addresses Report	LS_SIADDR	All
Service Request Activity - All	Service Request Activity - All	BIP Service Request Activity - All	srvreqaa	All
Service Request Activity - Public	Service Request Activity - Public	BIP Service Request Activity - Public	srvreqpa	All
Service Request Detail - Barcode	Service Request Detail	BIP Service Request Detail	srvreqdt	All
Signature Administration	Signature Administration	BIP Pharma Signature	LS_SIGNAD	All
Suspect List	HLS Suspect List	BIP HLS Suspect List	HLS_SUSPECT	All
Tables Report	Tables Report	XMLP Repository Tables - Current Query	rpostab	All
Transaction Processor Status Report	Transaction Processor Status Report	BIP Transaction Processor Status Report	TXNPROCSTAT	All
UT Account Detail	UT Account Detail	BIP UT Account Detail	cut_acdetail	All
eChannel - Fund Request Summary	eChannel - Fund Request Summary	BIP eChannel - Fund Request Summary	FUNDREQSUM	All
eChannel - Marketing Funds Detail	eChannel - Marketing Funds Detail	BIP eChannel - Marketing Funds Detail	mktfunddet	All
eChannel - Partner Profile	eChannel - Partner Profile	BIP eChannel - Partner Profile	partpro	All
eChannel - Partner Report Card	eChannel - Partner Report Card	BIP eChannel - Partner Report Card	part rptcard	All

Siebel Mobile/Sample Database Reports

Here's a list of Siebel Mobile app and sample database reports:

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
Account List	Account List	BIP Accounts - Current Query	ACLIST	Siebel Sales
Account Profile	CG Account Profile Report	BIP CG Account Profile Report	CS_ACCNTPRO	Siebel Consumer Goods
Account Promotion Deal Sheet	CG Promotion Plan Details	BIP CG Promotion Plan Details	CS_PROPLANDET	Siebel Consumer Goods

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
Account Service Profile	Account Service Profile	BIP Current Account Service Profile	ACSVCPRO	Core / Call Center
Actual Visits	Actual Visits	BIP Actual Visits	LS_ACTVIST	Siebel Life Sciences / Pharma
Admin Product Line List	Admin Product Line List	BIP Admin Product Line - Current Query	ADPRODLI	Core
Admin Sales Cycle - Current Query	Admin Sales Cycle - Current Query	BIP Admin Sales Cycle - Current Query	ADSLSCYC	Siebel Sales
Alphabetic Phone List	Contacts - Phone List	BIP Contacts - Alphabetic Phone List	CNTPHON	Core
BIP Expense Report	BIP Expense Report	BIP Expense Report	EXPREP	Core
By Opportunity	Contacts - By Opportunity	BIP Contacts - By Opportunity	CNTOPP	Core
By Sales Rep	Opportunity - Sales Rep	BIP Opportunities - By Sales Rep	OPSLSREP	Siebel Sales
Case Overview	PUB GOV Case Overview	BIP PUB GOV Case Overview	PUB_GOV_CASEOVRW	Siebel Public Sector
Clinical Site Followup Report	Clinical Site Followup Report	BIP Clinical Site Followup Report	LS_FOUPRE	Siebel Life Sciences / Pharma
Clinical Trip Report Without CRF	Clinical Trip Report Without CRF	BIP LS Clinical Trip Report Without CRF	LS_TRNCRF	Siebel Life Sciences / Pharma
Contact Current Query Personal	Contact current query personal	BIP Contact - Current Query(Personal)	CNTLISTPER	Siebel Sales
Contact List	Contact List	BIP Contacts - Current Query	CNTLIST	Core
Contacts Alphabetic Phone List Personal	Contacts Alphabetic Phone List Personal	BIP Contacts - Alphabetic Phone List(Personal)	CNTPHONPER	Siebel Sales
Current Agreement Detail	Current Agreement Detail	BIP Agreement Detail	AGDET	Core
Customer Invoice	FS - Customer Invoice	BIP FS - Customer Invoice	CUSTINV	Siebel Financial Services
CUT Trouble Ticket Activity - All	CUT Trouble Ticket Activity - All	BIP CUT Trouble Ticket Activity - All	SRVREQAA_TT_CUT	Siebel Communications
CUT Trouble Ticket Detail	CUT Trouble Ticket Detail	BIP CUT Trouble Ticket Detail	SRVREQDT_TT_CUT	Siebel Communications
CUT Trouble Ticket Summary	XMLP CUT Trouble Ticket Summary	BIP CUT Trouble Ticket Summary	SRVREQSM_TT_CUT	Siebel Communications
Disease Overview	HLS Disease Overview	BIP HLS Disease Overview	HLS_DISEASEOVRW	Siebel Public Sector
eChannel - Fund Request Summary	eChannel - Fund Request Summary	BIP eChannel - Fund Request Summary	FUNDREQSUM	Siebel PRM
eChannel - Marketing Funds Detail	eChannel - Marketing Funds Detail	BIP eChannel - Marketing Funds Detail	MKTFUNDDDET	Siebel PRM

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
eChannel - Partner Profile	eChannel - Partner Profile	BIP eChannel - Partner Profile	PARTPRO	Siebel PRM
eChannel - Partner Report Card	eChannel - Partner Report Card	BIP eChannel - Partner Report Card	PARTRPTCARD	Siebel PRM
EIM Interface Tables Report	EIM Interface Tables Report, list of EIM tables from the repository	XMLP Repository EIM Interface Tables - Current Query	RPOSEIM	Core / Tools
ESP Account Plan	Enterprise Selling Process	BIP ESP Account Plan Overview Report	ESPCOV	Siebel Sales
FINS Demo Health Providers by Specialty	FINS Demo Health Providers by Specialty - Healthcare	BIP FINS Demo Health Providers by Specialty - Healthcare	FD_HEALTH_PROVIDERS	Siebel Financial Services
FINS Investment Banking Client Profile	FINS Investment Banking Client Profile	BIP FINS Investment Banking Client Profile	FINCORP_CLIENTIB	Siebel Financial Services
FINS Life Policy - By City	FINS Life Policy - By City	BIP FINS Life Policy - By City	LIFEPOLCITY	Siebel Financial Services
FINS Life Policy - By Sales Rep	FINS Life Policy - By Sales Rep	BIP FINS Life Policy - By Sales Rep	LIFEPOLSLSRP	Siebel Financial Services
FINS Life Quotes - By Source	FINS Life Quotes - BY Source	BIP FINS Life Quotes - By Source	LIFEPOL_BY_SRC	Siebel Financial Services
FS Below Minimum Inventory Quantity Per Location	FS Below Minimum Inventory Quantity Per Location	BIP FS Below Minimum Inventory Quantity Per Location	BELOWMIN	Siebel Field Sales
FS Cycle Count Details	FS Cycle Count Details	BIP FS Cycle Count Detail	CYCCOUNTDET	Siebel Field Sales
FS Field Engineer Activity Summary	FS Field Engineer Activity Summary	BIP FS Field Engineer Activity Summary	FEASUM	Siebel Field Sales
FS Field Engineer Detail	FS Field Engineer Detail	BIP FS Field Engineer Detail	FEADET	Siebel Field Sales
FS Inventory Cost Detail	FS Inventory Cost Detail	BIP FS - Inventory Cost Detail	INCOSTDET	Siebel Field Sales
FS Pick Ticket Details	FS Pick Ticket Details	BIP FS Pick Ticket Details	PICKTICKET	Siebel Field Sales
FS Pick Ticket Details - No Barcode	FS Pick Ticket Details - No Barcode	BIP FS Pick Ticket Details - No Barcode	PICKTICKETNBC	Siebel Field Sales
FS Product List By Location	FS Product List By Location	BIP FS Product List By Location	PRODLISTLOC	Siebel Field Sales
FS Repair Detail	FS Repair Detail	BIP FS Repair Detail	REPDET	Siebel Field Sales
FS Repair Detail - No Barcode	FS Repair Detail - No Barcode	BIP FS Repair Detail - No Barcode	REPDETNBC	Siebel Field Sales
Group List	HLS Group List	BIP HLS Group List	HLS_GROUP	Siebel Public Sector
Incident List	PUB HLS Incident List	BIP PUB HLS Incident List	PUB_HLS_INCIDENT	Siebel Public Sector

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
INS All Policies for Contact	INS All Policies for Contact	BIP INS All Policies for Contact	INSALLPOLCON	Siebel Financial Services
INS Auto Policy Binder	INS Auto Policy Binder	BIP INS Auto Policy Binder	INSAPBND	Siebel Financial Services
INS Auto Policy Change Request	INS Auto Policy Change Request	BIP INS Auto Policy Change Request	INSAPCRQ	Siebel Financial Services
INS Auto Policy Memorandum	INS Auto Policy Memorandum	BIP INS Auto Policy Memorandum	INSAPMEM	Siebel Financial Services
INS Auto Policy Receipt of Payment	INS Auto Policy Receipt of Payment	BIP INS Auto Policy Receipt of Payment	INSAPROP	Siebel Financial Services
Investigative Case Overview	HLS Case Overview	BIP HLS Case Overview	HLS_CASEOVRW	Siebel Public Sector
Literature Fulfillment	Literature Fulfillment	BIP Literature Fulfillment	LITFUL	Siebel Sales
Loyalty Accrual Dispute Statement	Loyalty Accrual Dispute Statement	BIP LOY Accrual Dispute	ACCRUALDISPUTE	Siebel Loyalty
Loyalty Member Enrolment Statement	Loyalty Member Enrolment Statement	BIP LOY Member Enrolment	MEMENROLSTMT	Siebel Loyalty
Loyalty Member Statement	Loyalty Member Statement	BIP LOY Statement	MEMSTMT	Siebel Loyalty
Loyalty Membership Cancellation Statement	Loyalty Membership Cancellation Statement	BIP LOY Statement	MEMCANCSTMT	Siebel Loyalty
Loyalty Membership Renewal Statement	Loyalty Membership Renewal Statement	BIP LOY Statement	MEMRENEWALSTMT	Siebel Loyalty
Loyalty Partner Billing Statement	Loyalty Partner Billing Statement	BIP LOY Statement	PARTNERBILLINGSTMT	Siebel Loyalty
Loyalty Partner Statement	Loyalty Partner Statement	BIP LOY Partner Statement	PARTNERSTMT	Siebel Loyalty
Loyalty Point Purchase Statement	Loyalty Point Purchase Statement	BIP LOY Points Purchase Statement	MEMPURCPTSSTMT	Siebel Loyalty
Loyalty Points Adjustment Statement	Loyalty Member Points Adjustment Statement	BIP LOY Points Adjustment	POINTSADJUSTMENTSTMT	Siebel Loyalty
Loyalty Points Gift Statement	Loyalty Points Gift Statement	BIP LOY Points Purchase Statement	MEMGIFTPNTSSTMT	Siebel Loyalty
Loyalty Points Purchase Statement	Loyalty Points Purchase Statement	BIP LOY Points Purchase Statement	MEMPURCPTSSTMT	Siebel Loyalty
Loyalty Points Transfer Statement	Loyalty Points Transfer Statement	BIP LOY Points Purchase Statement	MEMTRNPTSSTMT	Siebel Loyalty
Loyalty Redemption Voucher Issue Statement	Loyalty Redemption Voucher Issue Statement		VOUCHERISSUESTMT	Siebel Loyalty
Loyalty SoM Assessment Statement	Loyalty Member Status Change Statement	BIP LOY SoM Assessment	SOMASSESSMENTSTMT	Siebel Loyalty
Loyalty Tier Change Statement	Loyalty Tier Change Statement	BIP LOY Tier Change	TIERCHANGESTMT	Siebel Loyalty

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
Loyalty Voucher Cancellation Statement	Loyalty Voucher Cancellation Statement	BIP LOY Voucher Statement	VOUCHERCANCSTMT	Siebel Loyalty
Loyalty Voucher Issue Statement	Loyalty Voucher Issue Statment	BIP LOY Voucher Statement	VOUCHERISSUESTMT	Siebel Loyalty
MDV	MDV Report	BIP LS Medical MDV Report	LS_MDV	Siebel Life Sciences / Pharma
Mobile User Summary Report	Mobile User Summary Report	BIP Mobile User Summary Report	MOBUSRSUM	Core
Mobile Users Status Report	Mobile Users Status Report	BIP Mobile Users Status Report	MOBUSRSTAT	Core
Opportunities - Summary	Opportunities - Summary	BIP Opportunities - Summary	OPSUM	Siebel Sales
Opportunity List	Opportunity List	BIP Opportunity List - Current Query	OPLIST	Siebel Sales
Order Detail	Order Detail	BIP Order Detail	ORDET	Core / Order Management
Pharma Account at a Glance	Pharma Account at a Glance	BIP Pharma Account at a Glance	LS_ACATGL	Siebel Life Sciences / Pharma
Pharma Call Frequency - Account	Pharma Call Frequency - Account	BIP Pharma Call Frequency - Account	LS_FREQAC	Siebel Life Sciences / Pharma
Pharma Call Frequency - Professional	Pharma Call Frequency - Professional	BIP Pharma Call Frequency - Professional	LS_FREQPF	Siebel Life Sciences / Pharma
Pharma Contracts	Pharma Contracts	BIP Pharma Contracts	LS_CONTRACT	Siebel Life Sciences / Pharma
Pharma Formulary List	Pharma Formulary List	BIP Pharma Formulary List	LS_FORMUL	Siebel Life Sciences / Pharma
Pharma Manager's Call Per Day	Pharma Manager's Call Per Day	BIP Pharma Manager's Call Per Day	LS_MGRPD	Siebel Life Sciences / Pharma
Pharma ME Event Agenda	Pharma ME Event Agenda	BIP Pharma ME Event Agenda	LS_MEEVAG	Siebel Life Sciences / Pharma
Pharma ME Event Invitee	Pharma ME Event Invitee	BIP Pharma ME Event Invitee	LS_MEEVIN	Siebel Life Sciences / Pharma
Pharma ME Event Professional Attended	Pharma ME Event Professional Attended	BIP Pharma ME Event Professional Attended	LS_MEEVAT	Siebel Life Sciences / Pharma
Pharma ME Event Sign-Ins	Pharma ME Event Sign-Ins	BIP Pharma ME Event Sign-Ins	LS_ETSIGN	Siebel Life Sciences / Pharma
Pharma ME Session Roster	Pharma ME Session Roster	BIP Pharma ME Session Roster	LS_MESERT	Siebel Life Sciences / Pharma
Pharma MedEd Profile	Pharma MedEd Profile	BIP Pharma MedEd Profile	LS_MEEDPF	Siebel Life Sciences / Pharma
Pharma Meeting Sign-In	Pharma Meeting Sign-In	BIP Pharma Meeting Sign-In	LS_MTSIGN	Siebel Life Sciences / Pharma

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
Pharma My Sample History	Pharma My Sample History	BIP Pharma My Sample History	LS_MYSAMP	Siebel Life Sciences / Pharma
Pharma Objectives	Pharma Objectives	BIP Pharma Objectives	LS_OBJECT	Siebel Life Sciences / Pharma
Pharma Orders Detail (eBusiness)	Pharma Orders Detail (eBusiness)	BIP Pharma Orders Detail (eBusiness)	PHORDET	Siebel Life Sciences / Pharma
Pharma Orders Summary (eBusiness)	Pharma Orders Summary (eBusiness)	BIP Pharma Orders Summary (eBusiness)	PHORSUM	Siebel Life Sciences / Pharma
Pharma Professional at a Glance	Pharma Professional at a Glance	BIP Pharma Professional at a Glance	LS_PROFGL	Siebel Life Sciences / Pharma
Pharma Professional List	Pharma Professional List	BIP Pharma Professional List	LS_PFLIST	Siebel Life Sciences / Pharma
Pharma Protocol Payments	Pharma Protocol Payments	BIP Pharma Protocol Payments	LS_PROTPAY	Siebel Life Sciences / Pharma
Pharma Sample Adjustment	Pharma Sample Adjustment	BIP Pharma Sample Adjustment	LS_SAMPADJ	Siebel Life Sciences / Pharma
Pharma Sample History	Pharma Sample History	BIP Pharma Sample History	LS_SAMPHIST	Siebel Life Sciences / Pharma
Pharma Sample Invoices	Pharma Sample Invoices	BIP Pharma Sample Invoices	LS_INVOICE	Siebel Life Sciences / Pharma
Pharma Sample Orders	Pharma Sample Orders	BIP Pharma Sample Orders	LS_SAMPORD	Siebel Life Sciences / Pharma
Pharma Sample Receipt Report	Pharma Sample Receipt Report	BIP Pharma Sample Receipt Report	LS_RECEIPT	Siebel Life Sciences / Pharma
Pharma Sample Transfer Out	Pharma Sample Transfer Out	BIP Pharma Sample Transfer Out	LS_TRANSOUT	Siebel Life Sciences / Pharma
Pipeline Report By Rep	Pipeline - By Rep	BIP Pipeline Report By Rep	PIPEREP	Siebel Sales
Planned vs Actual Patient Dates	Planned vs Actual Patient Dates	BIP Planned vs Actual Patient Dates	LS_PLNACT	Siebel Life Sciences / Pharma
Portfolio Map Report	Portfolio Map Report	BIP Portfolio Map Report	PORTMAP	Siebel Sales
Portfolio Plan Report	Portfolio Plan Report	BIP Portfolio Plan Report	PORTPLAN	Siebel Sales
Portfolio Selected Unit Analysis	Portfolio Selected Unit Analysis	BIP Portfolio Selected Unit Analysis	PORTSU	Core
Position List - Current Query	Position List - Current Query	BIP Position List - Current Query	POSLSTCQ	Core
PPR Membership Application	PPR Membership Application	BIP PPR Membership Application	APPLDET	Siebel PRM
Price Lists	Price Lists	BIP Price List - Current Query	PRICELST	Core / Order Management
Product Defect Detail	Product Defect Detail	BIP Product Defect Detail	PRODEFDT	Core / Order Management

Report Name	Description	Primary Integration Object Name	Template (.rtf)	Application
Protocol Sites Addresses Report	Protocol Sites Addresses Report	BIP Protocol Sites Addresses Report	LS_SIADDR	Siebel Life Sciences / Pharma
Service Request Activity - All	Service Request Activity - All	BIP Service Request Activity - All	SRVREQAA	Core / Call Center
Service Request Activity - Public	Service Request Activity - Public	BIP Service Request Activity - Public	SRVREQPA	Core / Call Center
Service Request Detail - Barcode	Service Request Detail	BIP Service Request Detail	SRVREQDT	Core / Call Center
Signature Administration	Signature Administration	BIP Pharma Signature	LS_SIGNAD	Siebel Life Sciences / Pharma
Suspect List	HLS Suspect List	BIP HLS Suspect List	HLS_SUSPECT	Siebel Public Sector
Tables Report	Tables report, list of tables from the repository	XMLP Repository Tables - Current Query	RPOSTAB	Core / Tools
Transaction Processor Status Report	Transaction Processor Status Report	BIP Transaction Processor Status Report	TXNPROCSTAT	Core
UT Account Detail	UT Account Detail	BIP UT Account Detail	CUT_ACDETAIL	Siebel Communications