## Contents

Preface .................................................................................................................................................. vii

  Audience ........................................................................................................................................ vii
  Documentation Accessibility ........................................................................................................ vii
  Related Documents .................................................................................................................. viii
  Conventions ............................................................................................................................. viii

1 Introduction

  Adapter Features ...................................................................................................................... 1-1
  The Siebel Application Model ............................................................................................... 1-4
  Integration with Siebel ........................................................................................................ 1-4
    Integrating with Siebel EAI Architecture ........................................................................ 1-5
  Using Application Explorer with Oracle Application Adapter for Siebel .................. 1-5
  BSE Versus Oracle Adapter J2CA Deployment ................................................................. 1-6

2 Configuring Oracle Application Server Adapter for Siebel

  Starting Application Explorer ............................................................................................... 2-2
  Configuring Repository Settings ........................................................................................ 2-3
  Creating a Repository Configuration .................................................................................... 2-3
    Creating a Configuration for BSE ..................................................................................... 2-3
    Creating a Configuration for J2CA .................................................................................. 2-4
    Connecting to a BSE or J2CA Configuration ................................................................... 2-6
  Establishing a Connection (Target) for Siebel ................................................................. 2-6
    Defining a Target to Siebel ............................................................................................... 2-7
    Connecting to a Defined Target ....................................................................................... 2-10
    Disconnecting From Siebel .............................................................................................. 2-10
    Editing a Target ............................................................................................................... 2-11
    Deleting a Target to Siebel ............................................................................................. 2-11
  Viewing Application System Objects ................................................................................... 2-11
    Viewing Metadata ............................................................................................................. 2-11
  Creating XML Schemas ........................................................................................................ 2-13
  Siebel Prerequisites for Working With Integration Objects ............................................. 2-16
  Creating Schemas for Siebel Integration Objects ............................................................. 2-17
  Creating Integration Object (IO) Nodes for Siebel .......................................................... 2-19
  Creating a Service Node for a Siebel Business Service .................................................. 2-23
  Creating and Testing a Web Service (BSE Configurations Only) .................................... 2-24
Creating a Web Service ................................................................. 2-25
Testing a Web Service................................................................. 2-26
Generating WSDL (J2CA Configurations Only).......................... 2-28
Configuring an Event Adapter ....................................................... 2-29
Creating and Modifying a Channel ............................................. 2-30

3 Oracle WebLogic Server Deployment and Integration
Adapter Integration with Oracle WebLogic Server .................. 3-1
Deployment of Adapter .............................................................. 3-1
Updating Adapter Configuration ................................................ 3-2

4 Integration With BPEL Service Components in the Oracle SOA Suite
Overview ....................................................................................... 4-1
Deployment of Adapter .............................................................. 4-1
Configuring a New Application Server Connection ................. 4-2
Designing an Outbound BPEL Process for Service Integration ... 4-7
  Siebel Service Integration ......................................................... 4-8
  Creating a New SOA Application for the Outbound BPEL Process 4-9
  Defining a BPEL Outbound Process ......................................... 4-14
  Deploying the BPEL Outbound Process .................................... 4-36
  Invoking the Input XML Document in the Oracle Enterprise Manager Console 4-40
  Testing Outbound BPEL and Mediator Processes ................... 4-43
Designing an Inbound BPEL Process for Event Integration ........ 4-43
  Generating WSDL for Event Integration .................................... 4-44
  Creating a New SOA Application for the Inbound BPEL Process 4-50
  Defining a BPEL Inbound Process ............................................. 4-54
  Deploying the BPEL Inbound Process ....................................... 4-63
  Triggering an Event in Siebel .................................................... 4-67

5 Integration With Mediator Service Components in the Oracle SOA Suite
Configuring a New Application Server Connection ................... 5-2
Configuring a Mediator Outbound Process .................................. 5-2
  Creating a New SOA Application for the Outbound Mediator Process 5-2
  Defining a Mediator Outbound Process ...................................... 5-6
  Deploying the Mediator Outbound Process ............................... 5-21
  Invoking the Input XML Document in the Oracle Enterprise Manager Console 5-25
Configuring a Mediator Inbound Process .................................. 5-28
  Creating a New SOA Application for the Inbound Mediator Process 5-28
  Defining a Mediator Inbound Process ....................................... 5-32

6 Integration With BPM Service Components in the Oracle SOA Suite
Overview ....................................................................................... 6-1
Deployment of Adapter .............................................................. 6-1
Configuring a New Application Server Connection ................... 6-2
Designing an Outbound BPM Process Using Transformations for Service Integration 6-2
  Creating an Empty Composite for BPM .................................... 6-2
Defining a BPM Outbound Process ............................................................ 6-4
Deploying the BPM Outbound Process ....................................................... 6-47
Invoking the Input XML Document in the Oracle Enterprise Manager Console 6-50
Designing an Inbound BPM Process Using Transformations for Event Integration 6-51
Creating an Empty Composite for BPM .................................................... 6-51
Defining a BPM Inbound Process ............................................................ 6-53
Deploying the BPM Inbound Process ....................................................... 6-86
Triggering an Event in Siebel ................................................................. 6-90

7 Configuring Outbound and Inbound Processing Using Oracle Service Bus

Overview of Application Adapter Integration with Oracle Service Bus ........... 7-1
Configuring Outbound Processing Using Oracle Service Bus (J2CA Configuration) 7-1
Starting Oracle Service Bus and Creating Project Folders ......................... 7-2
Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus 7-5
Publishing a WSDL From Application Explorer to Oracle Service Bus ........ 7-5
Configuring a File Type Business Service ............................................... 7-10
Configuring a Proxy Service .................................................................... 7-15
Configuring a Pipeline ............................................................................ 7-20
Configuring Inbound Processing Using Oracle Service Bus (J2CA Configuration) 7-29
Starting Oracle Service Bus and Creating Project Folders ......................... 7-29
Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus 7-31
Publishing a WSDL From Application Explorer to Oracle Service Bus ......... 7-32
Configuring a File Type Business Service ............................................... 7-41
Configuring a Pipeline ............................................................................ 7-45
Configuring Outbound Processing Using Oracle Service Bus (BSE Configuration) 7-50
Starting Oracle Service Bus and Creating Project Folders ......................... 7-50
Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus 7-53
Publishing a WSDL From Application Explorer to Oracle Service Bus ......... 7-53
Configuring a File Type Business Service ............................................... 7-56
Configuring a WSDL Type Business Service ......................................... 7-58
Configuring a Proxy Service .................................................................... 7-63
Configuring a Pipeline ............................................................................ 7-66

8 Troubleshooting and Error Messages

Troubleshooting ...................................................................................... 8-1
BSE Error Messages ................................................................................. 8-5
General Error Handling in BSE .............................................................. 8-5
Adapter-Specific Error Handling ............................................................ 8-6

9 Advanced User Tools

Web Services Policy-Based Security ....................................................... 9-1
Configuring Web Services Policy-Based Security ..................................... 9-2
Migrating Repositories ............................................................................ 9-9
A  Using Siebel Workflows

Overview...................................................................................................................................................  A-1
Siebel Workflows........................................................................................................................................  A-1
Using a Policy to Invoke a Siebel EAI Workflow .................................................................  A-2
Siebel Workflow - Outbound ........................................................................................................  A-2
Siebel Workflow - Inbound..............................................................................................................  A-3
Creating a Siebel Workflow ...............................................................................................................  A-3
  Creating a Siebel Workflow for an Event Using MQSeries Transport........................................  A-3
  Creating a Siebel Workflow for an Event Using File Transport............................................  A-8
  Creating a Siebel Workflow for an Event Using HTTP Transport .....................................  A-13
  Creating a Siebel Workflow for a Service Using MQSeries Transport.................................  A-16
  Creating a Siebel Workflow for a Service Using File Transport...........................................  A-21
  Creating a Siebel Workflow for a Service Using HTTP Transport ........................................  A-25

Glossary

Index
Preface

Oracle Fusion Middleware Application Adapter for Siebel User’s Guide for Oracle WebLogic Server describes how to provide connectivity and integrate with Siebel systems.

Audience


Documentation Accessibility

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.
Related Documents

For more information, see the following documents in the Oracle Enterprise Repository 11g Release 1 (11.1.1.3.0) documentation set:

- *Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Application Adapter Upgrade Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Application Adapter Best Practices Guide for Oracle WebLogic Server*
- Oracle’s Unified Method (OUM)

A wealth of additional Governance information can be found within Oracle’s Unified Method (OUM). OUM can be used by Oracle employees, Oracle Partner Network Certified Partners or Certified Advantage Partners, and Clients who either participate in the OUM Customer Program or are engaged on projects where Oracle provides consulting services. OUM is a web-deployed toolkit for planning, executing and controlling software development and implementation projects.

For more information about OUM, see the OUM FAQ at

http://my.oracle.com/portal/page/myo/ROOTCORNER/KNOWLEDGEAREASI/BUSINESS_PRACTICE/Methods/Learn_about_OUM.html

Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
Oracle WebLogic Server connects to a Siebel system through Oracle Application Adapter for Siebel. Oracle Application Adapter for Siebel provides connectivity and carries out interactions on a Siebel system. This chapter contains the following topics:

- Section 1.1, "Adapter Features"
- Section 1.2, "The Siebel Application Model"
- Section 1.3, "Integration with Siebel"
- Section 1.4, "Using Application Explorer with Oracle Application Adapter for Siebel"
- Section 1.5, "BSE Versus Oracle Adapter J2CA Deployment"

### 1.1 Adapter Features

Oracle Application Adapter for Siebel provides a means to exchange real-time business data between Siebel systems and other applications, databases, or external business partner systems. The adapter enables external applications for inbound and outbound processing with Siebel.

Oracle Application Adapter for Siebel can be deployed as a J2EE Connector Architecture (J2CA) version 1.0 resource adapter. This deployment is referred to as Oracle Adapter J2CA. It can also be deployed as a Web services servlet and as such is referred to as Oracle Adapter Business Services Engine (BSE).

Oracle Application Adapter for Siebel uses XML messages to enable non-Siebel applications to communicate and exchange transactions with Siebel using services and events. Services and events are defined as follows:

- **Services** (also known as outbound processing): Enables applications to initiate a Siebel business event.
- **Events** (also known as inbound processing): Enables applications to access Siebel data only when a Siebel business event occurs.

To support event functionality, channels are supported. A **channel** represents configured connections to particular instances of back-end or other types of systems. The channel is the adapter component that receives events in real time from the EIS application. The channel component can be a File reader, an HTTP listener, or an MQ listener. A channel is always EIS specific. The adapter supports multiple channels for a particular EIS, which enables the user to choose the optimal channel component based on deployment requirements.

Oracle Application Adapter for Siebel:
- Supports synchronous and asynchronous, bidirectional message interactions for Siebel Business Services, Business Components, and Integration Objects.
- Includes Oracle WebLogic Server Adapter Application Explorer (Application Explorer), a GUI tool that uses the Siebel Object Manager to explore Siebel metadata and build XML schemas or Web services.
- Supports Siebel transports—MQSeries, File, and HTTP. It also supports MSMQ messaging.
- XML schemas for Oracle Adapter J2CA.
- Web services for BSE.

Oracle Application Adapter for Siebel supports all 23 Siebel Industry Applications (SIA) through business objects, business components, business services, and integration objects. Siebel Industry Applications include industry verticals such as insurance, high technology, automotive, communications, media, financial services, life sciences, manufacturing, and consumer goods.

Siebel Industry Applications is tailored to the specific business requirements and processes of a particular industry with additional business logic in the form of business objects, business components, business services, and integration objects. Oracle Application Adapter for Siebel exposes and generates metadata and interacts with these industry-specific objects.

See Also: Oracle Application Server Adapter Concepts Guide

Oracle Adapter Business Services Engine (BSE) Architecture

Figure 1–1 shows the generic architecture for the Oracle Web service adapter for packaged applications. The adapter works with BSE, as deployed to a Web container in a J2EE application server. BSE serves as host to the adapters, enabling Web service requests to the adapters.

Application Explorer, a design-time tool deployed along with BSE, is used to configure adapter connections, browse EIS objects, and configure services. Metadata created while you perform these operations are stored in the repository by BSE.

BSE uses SOAP as a protocol for receiving requests from clients, interacting with the EIS, and sending responses from the EIS back to clients.
Oracle Adapter J2CA Generic Architecture

Figure 1–2 shows the generic architecture for the Oracle Adapter J2CA for packaged applications. The Oracle Adapter J2CA is deployed to a standard J2CA container and serves as host container to the adapters. The connector is configured with a repository.

Application Explorer, a design tool that works with the connector, is used to configure adapter connections, browse EIS objects, and configure services. Metadata created while you perform these operations are stored in the repository by the connector. The repository can be a file system or an Oracle database. It is deployed as a RAR file and has an associated deployment descriptor called ra.xml. You can create multiple connector factories by editing the Oracle WebLogic Server deployment descriptor ra.xml. For more information, see Chapter 3, "Oracle WebLogic Server Deployment and Integration".

Note: Do not use a file repository for BSE in production environments.
1.2 The Siebel Application Model

The Siebel Enterprise application defines a data abstraction layer that removes dependencies on the underlying database. It accomplishes this by using intermediate Business Components and Business Objects that represent database structures. A Business Component usually represents a table in a database. A Business Object is a group of related business components.

From a given business component, you can navigate the relationships defined for that component to another component. The path you use to traverse component relationships is called the navigation path. For example, if you want to obtain all addresses for a particular account, you can traverse the parent/child relationship between Account and Address to obtain those addresses. By using navigation paths, you can traverse nearly all of the business component relationships defined in the Siebel system.

In Siebel, Integration Objects are similar to Siebel Business Components but describe more complex hierarchical data relationships.

1.3 Integration with Siebel

You can use Oracle Application Adapter for Siebel to initiate a Siebel business process, such as add/update account, or you can use the adapter as part of an integration effort to connect Siebel and non-Siebel systems. Oracle Application Adapter for Siebel is bidirectional and can detect an event from Siebel by receiving a Siebel XML document emitted by Siebel.
When integrating with Siebel using Siebel XML documents, the adapter application developer must use existing Siebel Integration Objects or create new Siebel Integration Objects to use within a Siebel Workflow. The Workflow processes inbound or outbound Siebel XML and uses various transports such as MQSeries, File, and HTTP to exchange transactions with external systems. The Siebel Workflow is usually created by the Siebel administrator or developer using Siebel Workflow Administration screens.

When integrating with Siebel directly using the Java Data Bean or COM Data Interface, Oracle Application Adapter for Siebel does not require a Siebel Integration Object or Siebel Workflow. Instead, it executes Siebel Business Services and Siebel Business Components directly.

The following table lists Siebel objects and processes.

<table>
<thead>
<tr>
<th>Table 1–1 Siebel Objects and Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siebel Objects</td>
</tr>
<tr>
<td>Business Services</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Business Components</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Integration Objects</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

1.3.1 Integrating with Siebel EAI Architecture

Siebel enables integration with other applications and systems using its Siebel EAI (Enterprise Application Integration) framework and its Business Integration Manager facility. Oracle Application Adapter for Siebel uses the Siebel EAI framework and leverages various integration access methods to provide the greatest amount of flexibility and functionality while working within the Siebel framework.

Oracle Application Adapter for Siebel supports the following integration access methods:

- Siebel Java Data Bean for services involving Siebel Business Components or Siebel Business Services.
- Siebel COM Data Interface for services involving Siebel Business Components or Siebel Business Services.
- Siebel XML for events and services involving Siebel Integration Objects.

1.4 Using Application Explorer with Oracle Application Adapter for Siebel

Application Explorer uses an explorer metaphor for browsing the Siebel system for Business Services, Business Objects, Business Components, and Integration Objects. The explorer enables you to create XML schemas and Web services for the associated object. External applications that access Siebel through Oracle Application Adapter for Siebel use either XML schemas or Web services to pass data between the external application and the adapter.
Application Explorer uses interfaces provided by Siebel and in-depth knowledge of the Siebel application systems to access and browse business object metadata. After an object is selected, Application Explorer can generate an XML schema or Web service to define the object for use with Oracle Application Adapter for Siebel.

Key features of Application Explorer include:

- The ability to connect to and explore a variety of application systems.
- Access to application system object metadata.
- A point-and-click process for generating XML schemas and Web services.

See Also:

- Oracle Application Server Adapter Concepts Guide
- Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server

1.5 BSE Versus Oracle Adapter J2CA Deployment

If you are using Oracle Application Adapter for Siebel with Oracle SOA Suite components (for example, BPEL, Mediator, BPM, or OSB), then note that:

- Only Oracle Adapter J2CA deployment supports inbound integration (event notification) with Oracle SOA Suite components.
- Oracle Adapter J2CA and BSE deployments support outbound integration (request-response service) with Oracle SOA Suite components.

The following two factors explain the differences between deploying BSE and Oracle Adapter J2CA. Understanding these factors can help in selecting a deployment option.

1. BSE has the following advantages:

- Can be deployed in a separate instance of Oracle WebLogic Server.
- Provides better distribution of load.
- Conforms more closely to the Service Oriented Architecture (SOA) model for building applications.

2. Oracle Adapter J2CA does provide slightly better performance than BSE.
This chapter describes how to configure Oracle Application Adapter for Siebel and create schemas for Siebel Business Objects. It contains the following topics:

- Section 2.1, "Starting Application Explorer"
- Section 2.2, "Configuring Repository Settings"
- Section 2.3, "Creating a Repository Configuration"
- Section 2.4, "Establishing a Connection (Target) for Siebel"
- Section 2.5, "Viewing Application System Objects"
- Section 2.6, "Creating XML Schemas"
- Section 2.7, "Siebel Prerequisites for Working With Integration Objects"
- Section 2.8, "Creating Schemas for Siebel Integration Objects"
- Section 2.9, "Creating Integration Object (IO) Nodes for Siebel"
- Section 2.10, "Creating a Service Node for a Siebel Business Service"
- Section 2.11, "Creating and Testing a Web Service (BSE Configurations Only)"
- Section 2.12, "Generating WSDL (J2CA Configurations Only)"
- Section 2.13, "Configuring an Event Adapter"

**Encoding Support on UNIX Platforms**

**Important (All UNIX Platforms):** Before you attempt to connect to a Siebel target using a BSE or J2CA configuration in a UNIX environment, you must perform the additional steps described in "Adding Required Encoding Option (All UNIX Platforms)" on page 2-1. Failure to add the encoding option as described in this section results in an error and prevents you from connecting to the Siebel target. The error message may indicate that the encoding is not supported, for example:

```
Error: Problem activating adapter -- UTF-8 is not supported. Check logs for more information.
```

```
Error: Error getting target [Siebel] -- UTF-8 is not supported.
```

**Adding Required Encoding Option (All UNIX Platforms)**

Before attempting to connect to a Siebel target, add the following Java file encoding option to the startWebLogic.sh file:

```
JAVA_OPTIONS="${SAVE_JAVA_OPTIONS} -Dfile.encoding=ISO8859_1"
```
The startWebLogic.sh file is located in the following directory:

<ADAPTER_HOME>/user_projects/domains/base_domain/lib

**Siebel Connectivity Prerequisites for Versions 6.2 and Lower**

For Siebel versions 6.2 and lower only, you must perform the following steps to connect to your Siebel system using COM connectivity for a J2CA configuration.

1. Install Siebel thick client on the same system where the adapters are installed.
2. Install the database client (Microsoft SQL Server or Oracle) on the same system.
3. The Siebel .DLL files (iwsiebel.local.dll and iwsiebel.core.dll) in the adapter lib folder must be added to the Application server path.
4. Edit the uagent.cfg file and change the data source parameter value from "local" to "server".
   The uagent.cfg file can be found in the following Siebel thick client folder:
   c:\sea\client\bin
5. Edit the data source for SEA MSQl with appropriate parameters.
   You can edit a data source in Windows by accessing the Control Panel, Administrative Tools, and Data Sources (ODBC).
6. Use the following target type when creating the adapter target connection:
   Siebel 6.2 - (Local COM Access Implementation)
7. Provide the full path to the uagent.cfg file when creating an adapter target connection, for example:
   c:\sea\client\bin\uagent.cfg

### 2.1 Starting Application Explorer

To start Application Explorer:

1. Ensure that Oracle WebLogic Server is started, which is where Application Explorer is deployed.
2. Click the Windows Start menu, select All Programs, Oracle Application Adapters, and then click Application Explorer, as shown in Figure 2–1.

**Figure 2–1 Application Explorer Option**

Application Explorer is displayed.

Alternatively, you can run the ae.bat file, which is located in the following directory to start Application Explorer:

Oracle SOA Suite:

<ORACLE_HOME>/soa/thirdparty/ApplicationAdapters/tools/iwae/bin/ae.bat
Oracle Service Bus (OSB):

<OSB_HOME>\Oracle_OSB1\3rdparty\ApplicationAdapters\tools\iwae\bin\ae.bat

It is a good practice to create a shortcut for the ae.bat file on your desktop.

If you are using a UNIX or Linux platform, then you can start Application Explorer by using the iwae.sh file.

Application Explorer starts. You are ready to define new targets to your Siebel system.

---

**Note:** Before you run the iwae.sh file on UNIX or Linux platforms, the permissions must be changed. For example:

```bash
chmod +x iwae.sh
```

---

### 2.2 Configuring Repository Settings

A repository holds information about configuration details, adapter targets, channels, and other configuration information. For more information on how to configure BSE and J2CA repository settings, see the *Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server* (Section 2.8.4, "Configuring the Database Repository").

### 2.3 Creating a Repository Configuration

Before you use Application Explorer with Oracle Application Adapter for Siebel, you must create a repository configuration. You can create two kinds of repository configurations, Web services and J2CA, depending on the container to which the adapter is deployed.

During design time, the repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at run-time.

Web services and BSE refer to the same type of deployment. For more information, see "Adapter Features" on page 1-1.

#### 2.3.1 Creating a Configuration for BSE

To create a configuration for BSE using Application Explorer, you must first define a new configuration.

**Defining a New Configuration for BSE**

To create a new configuration for BSE:

1. Start the Application Explorer.
2. Right-click **Configurations** and select **New**.
   
   The New Configuration dialog is displayed.
3. Enter a name for the new configuration (for example, SampleConfig) and click **OK**.
   
   The New Configuration dialog is displayed, as shown in Figure 2–2.
Creating a Repository Configuration

**Figure 2–2  New Configuration Dialog**

<table>
<thead>
<tr>
<th>Service Provider: iBSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>iBSE URL: <a href="http://localhost:7001/ibse/IBSEServlet">http://localhost:7001/ibse/IBSEServlet</a></td>
</tr>
</tbody>
</table>

4. From the **Service Provider** list, select **iBSE**.

5. In the **iBSE URL** field, accept the default URL or replace it with a different URL using the following format:

   http://host name:port/ibse/IBSEServlet

   Where **host name** is the system where your Oracle WebLogic Server resides and **port** is the HTTP port number on which the Oracle WebLogic Server is listening.

6. Click **OK**.

   As shown in **Figure 2–3**, a node representing the new configuration appears beneath the root Configurations node.

**Figure 2–3  SampleConfig Node**

<table>
<thead>
<tr>
<th>Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>SampleConfig</td>
</tr>
</tbody>
</table>

### 2.3.2 Creating a Configuration for J2CA

To create a configuration for Oracle Adapter J2CA using Application Explorer, you must first define a new configuration.

#### Defining a New Configuration for J2CA

To define a new configuration for J2CA:

1. Start the Application Explorer.

2. Right-click **Configurations** and select New, as shown in **Figure 2–4**.

**Figure 2–4  Configurations Node**

<table>
<thead>
<tr>
<th>Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibs New</td>
</tr>
</tbody>
</table>

The New Configuration dialog is displayed, as shown in **Figure 2–5**.

3. Enter a name for the new configuration (for example, SampleConfig) and click **OK**.
Creating a Repository Configuration

Configuring Oracle Application Server Adapter for Siebel

Figure 2–5  New Configuration Dialog

4. From the Service Provider list, select JCA.

5. Click OK.

As shown in Figure 2–6, a node representing the new configuration appears beneath the root Configurations node.

Figure 2–6  SampleConfig Node

The Oracle Adapter J2CA configuration folder is stored in:

<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\config\configuration_name

Where configuration_name is the name of the configuration you created (for example, myConfig).

HTTP Repository Connection

J2CA users can create an HTTP repository connection, which enables them to generate and store WSDL documents remotely. Perform the following steps to create an HTTP repository connection in Application Explorer. To use the HTTP repository, make sure that the iwjcaivp test tool(jca-app-adapter-test) is successfully deployed and running.

1. Start the Application Explorer.

2. Right-click the Configurations node in the left pane and select New.

   The New Configuration dialog opens.

3. Type a name for the configuration and click OK.

4. Select JCA from the Service Provider list box and enter an HTTP target value in the Home field.

   Use the following format for the HTTP target value:

   http://host name:port/iwafjca/JCAServlet

   For example:

   http://iwserv14:8001/iwafjca/JCAServlet

5. Click OK.

The new HTTP repository connection is added to the Configurations node.

Once you connect to the remote server, you can create new Adapter targets, generate WSDL documents, and store them in the remote server.
Note: When you configure an Adapter target with the J2CA HTTP repository, you are not required to restart the Oracle WebLogic Server for run time purposes.

2.3.3 Connecting to a BSE or J2CA Configuration

To connect to a new configuration:

1. Right-click the configuration to which you want to connect, for example, SampleConfig.
2. Select Connect.

Nodes appear for Adapters, Events, and Business Services (also known as Web services). The Business Services node is only available for BSE configurations. If you are connected to a J2CA configuration, then the Business Services node is not shown. As shown in Figure 2–7, the following is an example of a BSE configuration named SampleConfig:

![Figure 2–7](Image)

- Use the Adapters folder to create inbound interaction with Siebel. For example, you use the Siebel node in the Adapters folder to configure a service that updates Siebel.
- Use the Events folder (available for J2CA configurations only) to configure listeners that listen for events in Siebel.
- Use the Business Services folder (available for BSE configurations only) to test Web services created in the Adapters folder. You can also control security settings for the Web services by using the security features of the Business Services folder.

You can now define new targets to Siebel.

2.4 Establishing a Connection (Target) for Siebel

To browse the Siebel Business Services, Business Components, and Integration Objects, you must define a target to Siebel. After you define the target, the parameters are automatically saved.

**Important (All UNIX Platforms):** Before you attempt to connect to a Siebel target using a BSE or J2CA configuration in a UNIX environment, you must perform the additional steps described in "Adding Required Encoding Option (All UNIX Platforms)" on page 2-7. Failure to add the encoding option as described in this section results in an error and you are not able to connect to the Siebel target. The error message may indicate that the encoding is not supported, for example:

Error: Problem activating adapter -- UTF-8 is not supported. Check logs for more information.

Error: Error getting target [Siebel] -- UTF-8 is not supported.
Adding Required Encoding Option (All UNIX Platforms)
Before attempting to connect to a Siebel target, perform the following steps:

1. Add the following Java file encoding option to the `startWebLogic.sh` file:
   ```
   JAVA_OPTIONS="${SAVE_JAVA_OPTIONS} -Dfile.encoding=ISO8859_1"
   ```

   The `startWebLogic.sh` file is located in the following directory:
   ```
   <ADAPTER_HOME>/user_projects/domains/base_domain/bin
   ```

2. Log in to the Oracle WebLogic Server console, click Administration, then Server Properties, and under Command Line Options, edit the Java Options field to include the following:
   ```
   -Dfile.encoding="ISO8859_1"
   ```

2.4.1 Defining a Target to Siebel

The connection parameters required for defining a Siebel target can be obtained from the `eapps.cfg` file, which is located in the following directory:

```
drive:\SiebelRoot\SWEApp\BIN
```  

Where Siebelroot is the Siebel installation directory.

When you are working with a J2CA configuration, creating, updating, and deleting a target requires you to restart the Oracle WebLogic Server. The Oracle WebLogic Server must also be restarted after you create a target, connect to a target, and generate a WSDL for a Siebel object. In addition, make sure to close Application Explorer before you restart the Oracle WebLogic Server.

To define a target to Siebel:

1. In the left pane, expand the Adapters node, as shown in Figure 2–8.

   ![Adapters Node](image)

2. Right-click the Siebel node and select Add Target.

   The Add Target dialog is displayed. Provide the following information:
   a. In the Name field, enter a name for the new target.
   b. In the Description field, enter a description (optional).
   c. From the Target Type list, select Java Bean Data Connection (default).

3. Click OK.

   The Java Data Bean Connection dialog is displayed, as shown in Figure 2–9.
Enter the system information as specified in the following steps:

a. In the Gateway Server field, enter the name of the server. To specify a Gateway Server that uses a port other than the default (usually, 2320), add a colon and the port number, for example, `gateway name:port number`.

b. In the Enterprise Name field, enter the appropriate name.

c. In the Siebel Server field, enter the name of your Siebel server. Do not supply a value in this field when connecting to a Siebel 7.7, 7.8, or 8 system.

d. In the User field, enter the user name.

e. In the Password field, enter the password associated with the user name.

f. From the Siebel Version list, select **Siebel 7.7 and above** (default) or **Siebel 7.5 and below**.

g. Click the Advanced tab, as shown in Figure 2–10 and verify the following:
   - Language
   - Object Manager
For Siebel 7.0.3, the default Object Manager is EAIObjMgr. For Siebel 7.7, the default is EAIObjMgr_enu. Siebel 7.7 requires that you add a language extension (for example, _enu) to the end of the Object Manager name. Check with your Siebel Administrator for the specific names that apply to your system.

**Repository Name**

If no repository is specified, then a full list of objects from all available repositories is returned. If a specified repository is not found, then an empty list of objects is returned.

The configuration parameters supplied are those used by Siebel client applications to connect to the Siebel system. For more information about these parameters, see your Siebel documentation or ask your Siebel system administrator.

**Encryption**

A new parameter named Encryption is now introduced to the Advanced tab when using the Siebel adapter to create a target during design time. This parameter has two values, None and RSA. The default value is None, where no encryption is performed. By choosing RSA, an RSA-encrypted connection to the object manager specified is established.

To use RSA encryption, the Object Manager must be specified as SCCObjMgr_enu.
4. Click OK.

In the left pane, the target you create appears under the Siebel node.

2.4.2 Connecting to a Defined Target

To connect to a defined target:

1. Expand the Siebel node and click the target name to which you want to connect, as shown in Figure 2–11.

Figure 2–11 Disconnected Siebel Target

![Disconnected Siebel Target](image)

2. In the left pane, right-click the target name and select Connect.

The target icon changes, indicating that you are connected to the Siebel system, as shown in Figure 2–12.

Figure 2–12 Siebel Target Node

![Siebel Target Node](image)

You can now browse the available Business Objects, Business Services, and Integration Objects in the Siebel system.

2.4.3 Disconnecting From Siebel

Although you can maintain multiple open connections to different application systems, it is good practice to close connections when not in use.

To disconnect from Siebel:

1. In the left pane, select the target to which you are connected.

2. Right-click the target and select Disconnect.

Disconnecting from the application system drops the target, but the node remains. The SiebelConnection node in the left pane changes to reflect that the target is disconnected, as shown in Figure 2–13.

---

Note: These parameters are typically found in Siebel configuration files stored under the Siebel server root/bin/<language> directory, where language is the Siebel code for the language you installed (enu for U.S English). For example, for Siebel versions 7 and higher on a Windows platform, for the Siebel Call Center module, these values can be found in the uagent.cfg file. Consult your Siebel administrator and your Siebel bookshelf documentation for more information.
2.4.4 Editing a Target

To edit a target:

1. In the left pane, ensure the target you want to edit is disconnected.
2. Right-click the disconnected target and select **Edit**, as shown in Figure 2–14.

![Figure 2–13  Disconnected Siebel Target](image)

**Figure 2–14  Edit Option**

![Figure 2–14  Edit Option](image)

The Edit pane is displayed on the right.

3. Modify the target information.
4. Click **OK**.

2.4.5 Deleting a Target to Siebel

You can delete a target, rather than just disconnecting and closing it. When you delete the target, the node disappears from the list of Siebel targets in the left pane of Application Explorer.

When you delete a target, you must restart the Oracle WebLogic Server to update the repository for run time purposes.

To delete a target:

1. In the left pane, select the target.
2. Right-click the target and select **Delete**.
   
   A confirmation message is displayed.
3. Click **OK** to delete the target you selected.
   
   The Siebel connection node disappears from the left pane.

2.5 Viewing Application System Objects

Application Explorer gives you the flexibility to view all Siebel application system objects. One benefit of this flexibility is that you can gain an understanding of the Siebel data structure. You can review parameters, data types, and other attributes of the Siebel data in the right pane.

2.5.1 Viewing Metadata

To view metadata:

1. Start Application Explorer and connect to your Siebel system.
2. In the left pane, expand the Business Object or Business Service containing the component for which you want to generate schema.

3. Expand the Business Object or Business Service node.

4. Expand the Business Component or the Business Service node to view the objects under it.
   - For a Business Component, select the node in which you are interested, for example, Account, as shown in Figure 2–15.

   **Figure 2–15  Account Node**
   ![Account Node](image)
   - For a Siebel Business Service, select the object in which you are interested, for example, addAccount, as shown in Figure 2–16.

   **Figure 2–16  Simple Add Account Node**
   ![Simple Add Account Node](image)

5. In the right pane, click the ellipsis (...) in the Table row of the properties table.

   The metadata table appears in the right pane, as shown in Figure 2–17.

   **Figure 2–17  Metadata Table for the Siebel Object**
   ![Metadata Table](image)
2.6 Creating XML Schemas

You can create service schemas for Business Services and Business Components using Application Explorer.

The following topic describes how to create schemas for the adapter when you deploy Oracle Application Adapter for Siebel for use either in a J2CA environment or a Web services environment. For more information, see "Creating and Testing a Web Service (BSE Configurations Only)" on page 2-24 if you plan to deploy Oracle Application Adapter for Siebel in a Web services environment.

Siebel Schema Considerations

When inserting a record into Siebel, the data can be specified by the user or configured in Siebel to have default values or other system generated values. For example the Account Business Component, Currency Code, by default, has ‘USD’ and the system fields such as ROW_ID generated by the Siebel system when the record is inserted. The Siebel API does not provide this distinction. Therefore, the Oracle Application Adapter for Siebel cannot anticipate what the required fields the user should enter are and what are the required fields that can be filled by Siebel. As a result, the adapter schemas have been modified to have all elements as optional by setting minoccurs=0 for the elements.

Hence, all users must determine which fields are mandatory through Siebel Tools and create a payload (request XML document) for Siebel services (outbound).

Creating an XML Schema for a Siebel Business Object or Business Service

You create schemas for Siebel Business Service methods (for example, the Add method) and Business Components using Application Explorer. After you create a schema, you can use it to generate service request and response schemas for the Business Service or Business Component.

Siebel Business Objects contain one or more Siebel Business Components. You can view Business Components by clicking the associated Business Object.

For example, the Account Business Object can be expanded to display all available Business Components, as shown in Figure 2–18.
Creating an XML Schema for a Siebel Business Component or Business Service

To generate service request and response schemas for a Business Component or Business Service:

1. Start Application Explorer and connect to your Siebel system.
2. In the left pane, expand the Business Object or the Business Service node.
3. Expand the Business Component or Business Service to view the objects under it.
   - For a Business Component, expand the Business Object node, then expand the Business Component you want, then expand the node you want, and select the method for which you want to create a schema, as shown in Figure 2–19.
   - For a Siebel Business Service, expand the Business Service node containing the object for which you want to create schema, as shown in Figure 2–20.
4. Right-click the node and select Generate Schema.
   Application Explorer accesses the Siebel repository and builds schemas.
As shown in Figure 2–21, schema tabs similar to the following appear in the right pane.

![Request and Response Schema Tabs](image)

5. To view a schema, click the ellipsis tab corresponding to the schema you want to view.

The schema appears on the right, as shown in Figure 2–22.

![XML Schema](image)

**Searching for a Specific Siebel Object**

You can use the search function in Application Explorer to locate a Siebel object or node quickly.

1. Start Application Explorer and connect to your Siebel system through a target.
2. Expand the target and select Business Object, Business Service, or Integration Object.
3. In the right pane, move the cursor over Operations and select Search.
4. Enter the name of the node or object on which you want to search in the text entry box, for example, Account.
5. Click OK.

A list containing the Siebel items that match your search appears.

6. Select the item in which you are interested.

Application Explorer locates the item in which you are interested.

**Returning Fields in a Specified Order**

When you create a request document from an XML schema to query the Siebel system, you can limit the expected response to specific fields that are specified in the query.
The response contains the fields in the order in which they were specified. If you do not specify a set of fields, then the response document contains the entire set.

For example, the following query returns all fields:

```
<m:Siebel location="S/BO/Account/Account/queryWithView" view="AllView">
   <m:select>
      <m:Name>Yelena*</m:Name>
   </m:select>
</m:Siebel>
```

The following query returns a response that only contains the fields Name, Location and Account Status fields:

```
<m:Siebel location="S/BO/Account/Account/queryWithView" view="AllView">
   <m:select>
      <m:Name>Yelena*</m:Name>
   </m:select>
   <m:field>Name</m:field>
   <m:field>Location</m:field>
   <m:field>Account Status</m:field>
</m:Siebel>
```

**Using QueryWithView**

For Business Components, the iWay Application Adapter for Siebel enables Insert, Update, Delete, and Query. It also enables a method called QueryWithView. The View modes are a visibility feature provided by Siebel.

By using QueryWithView, you can specify a Siebel View mode as a parameter. The API parameters allow different presentations of data depending on the Siebel environment that you configured.

You can use Query except when you want to enable a user to retrieve records based on different view modes. In this case, use QueryWithView. For more information on QueryWithView mode or Siebel "Visibility" concepts, see your Siebel Administrator.

The following levels are available:

- Sales Rep View
- Manager View
- Personal View
- All View
- Organization View
- Group View
- Catalog View
- SubOrganization View

**2.7 Siebel Prerequisites for Working With Integration Objects**

To create XML schemas for Siebel Integration Objects, you may have to generate XDR schemas first, using the Siebel Tools Schema Wizard.

The XDR schema is used as input to Application Explorer when generating schemas for integration objects. After you generate the XDR schema, Application Explorer uses the XDR file to generate the XML schema.

Please note:
For **Siebel 7.5 and later**: Generate XSD schemas directly from Siebel tools. These XSD schemas are used to create Web services directly using Application Explorer. After you generate an XSD schema through Siebel tools, use it to create an IO node and Web service.

For **Siebel 7.0**: You cannot generate XSD schemas directly from Siebel tools; only XDR schemas can be created. Therefore, to create a Web service, Application Explorer must first generate an XSD schema from the XDR schema.

For releases **before Siebel 6.3**: The Siebel Tools Schema Wizard creates only DTD schemas. You must transform these schemas manually, or by using other tools, into XDR files before Application Explorer can use them as input to create XML schemas. In addition, you must include the SiebelMessage tag reference in your XDR file.

Oracle Application Adapter for Siebel supports access to Siebel Integration Objects by using Siebel XML to handle events. Using Siebel Integration Objects through supported transports requires Siebel workflows.

### 2.8 Creating Schemas for Siebel Integration Objects

This section describes how to create schemas for Siebel Integration Objects.

**Creating a Siebel XDR or XSD Schema for a Siebel Integration Object**

To generate a Siebel XDR or XSD schema:

1. Log on to Siebel Tools, as shown in Figure 2–23.

**Figure 2–23 Siebel Tools Menu**

Perform the following steps:

a. Enter your user ID and password.

b. Select a database from the list.

2. Click OK.
The Siebel Tools window is displayed, as shown in Figure 2–24. Integration Objects appear in the right pane.

**Figure 2–24  Siebel Tools Window**

3. To create a schema, select an Integration Object, for example, Sample Account.
4. Click **Generate Schema**.

   The Generate XML Schema wizard is displayed, as shown in Figure 2–25.

**Figure 2–25  Generate XML Schema Wizard**

Perform the following steps:
a. From the Select a Business Service list, select **EAI XML XDR Generator** for XDR schemas or **EAI XML XSD Generator** for XSD schemas (for Siebel 7.5 and later).

b. From the Select an envelope type list, select **Siebel Message envelope**.

c. In the Choose the file name field, specify a file name for the XDR schema and a directory where it can be accessed by Application Explorer.

**Note:** The XDR or XSD schema file must be saved to a directory on the same computer as Application Explorer.

5. Click **Finish**.

6. Create a workflow to accept incoming XML documents through HTTP and to insert/update Siebel data by using the EAI XML Converter and EAI Siebel Adapter Business Services.

For more information, see Appendix A, "Using Siebel Workflows".

7. Edit the **eai.cfg** file, which is located in the following directory:

```xml
<siebel_server>/bin/enu
```

8. Add the following line to the `[HTTP Services]` section:

```plaintext
[HTTP Services]
wf = iWayWorkflow
```

9. Confirm that the following line is set in the `[EAI_ENU]` section of the `Eapps.cfg` file:

```plaintext
[EAI_ENU]
EnableExtServiceOnly = True
```

The `Eapps.cfg` file is located in the following directory:

```xml
<siebel_server>/bin
```

10. Create a named subsystem using Siebel Server Manager by running the following command, where EAITEST is the name of the workflow that was created in step 6:

```plaintext
create named subsystem iWayWorkflow for subsystem EAITransportDataHandlingSubsys with DispatchWorkflowProcess="EAITEST"
```

Now you can use Application Explorer to create Integration Object (IO) nodes for Siebel.

### 2.9 Creating Integration Object (IO) Nodes for Siebel

To create an Integration Object node for Siebel, perform the following steps:

1. In Application Explorer, connect to a defined target. For more information on how to connect to a target, see "Connecting to a Defined Target" on page 2-10.

The X over the icon disappears, indicating that the node target is connected, as shown in Figure 2–26.
2. Expand the Integration Object node and select Sample Account.

3. Right-click the **Sample Account** node and select **Add IO Node**.

   The Add IO Node dialog is displayed, as shown in Figure 2–27.

---

**Figure 2–26 Expanded Siebel Node**

```
Siebel
  siebel_target
    Business Object
    Business Service
    Integration Object
```

---

**Figure 2–27 Add IO Node Dialog**

---

Please note:

- **For Siebel 7.5 or later:** Generate XSD schemas directly from Siebel tools. You use the XSD schemas when you create Web services in Application Explorer. After you generate an XSD schema through Siebel tools, use it to create an IO node and a Web service.

- **For Siebel 7.0:** You cannot generate XSD schemas directly from Siebel tools; only XDR schemas can be created. Before you create a Web service, you must first generate an XSD schema from the XDR schema using Application Explorer.

---

**Note:** This is the schema file that you generated in **Creating Schemas for Siebel Integration Objects** on page 2-17.

---

4. Enter a node name, for example SampleAccount in the **Node name** field and a path to the Sample Account XDR or XSD file in the **Schema location** field.

5. If the XSD schema has already been generated, then select XSD Schema. If you are using Siebel-generated XDR schemas, then do not select the XSD schema option.

6. Select a protocol from the **Protocol** list.

7. Click **Continue**.

   The Add IO Node dialog is displayed, as shown in Figure 2–28.
Figure 2–28  Add IO Node Dialog

8. Perform the following steps:
   a. In the SWE URL field, type the Base SWE URL. For example:
      
      http://web_server/eai/start.swe
      Where web_server is the name of the Web server that is hosting Siebel SWE.
   b. In the SWE External Source field, type the section within the eai.cfg file to execute, which is the [HTTP Services] section.
      
      For more information, see step 8 in Creating Schemas for Siebel Integration Objects on page 2-17.
   c. In the SWE External Command field, type the following command exactly as shown:
      
      Execute
   d. In the User Name and Password fields, type a valid user name and password used to connect to the Siebel SWE.
      
      The user name and password must have privileges to execute the given workflow.
9. Click Finish.
   
   The new IO node is listed under the Integration Object's Sample Account node, as shown in Figure 2–29.

Figure 2–29  Integration Object's Sample Account Node

   You can now create an XML schema.

Creating an XML Schema for a Siebel Integration Object

After you create an Integration Object node for Siebel, you can create an XML schema using Application Explorer.
To create an XML schema:

1. In Application Explorer, expand the **Integration Objects** node to browse the Integration Objects in the Siebel system, as shown in **Figure 2–30**.

![Siebel Integration Objects Node, Sample Account](image)

2. Scroll down and select an Integration Object (for example, SampleAccount).

3. Right-click the created Integration Object node (for example, SampleAccount) and select **Export Schema(s)** from the menu, as shown in **Figure 2–31**.

![Export Schema(s) Menu Option](image)

The Select Export Directory dialog is displayed, as shown in **Figure 2–32**.
The exported event schema must be specified during the channel creation process in the PreParser tab (Schema location field).

2.10 Creating a Service Node for a Siebel Business Service

OracleAS Adapter for Siebel enables the addition of a service node for a Business Service that includes methods containing method arguments having hierarchy data types.

Important limitations:

- The adapter supports only Integration Object hierarchy data types.
- Adding a Service node requires that you have previously generated an XSD schema for the Integration Object. For more information on generating XSD schemas for Siebel Integration Objects, see "Creating Schemas for Siebel Integration Objects" on page 2-17.
- Only one of the method arguments for the Business Service method for which you want to add a service node can be a hierarchical data type.
- The method argument XMLCharEncoding is not supported. Leave this element blank in the XML payload. If you enter a valid XMLCharEncoding value such as UTF-8 or UTF-16, then the following error is received:

  Invocation of Service failed.

To create the service:

1. Select the Business Service node in which you are interested.

2. Right-click the Business Service method argument for which you want to create a service and select Add Service Node.

The Add Service Node dialog is displayed, as shown in Figure 2–33.
3. Perform the following steps:
   a. Provide a service node name.
   b. Enter a description (optional).
   c. Provide the full path (including the file name) to the XSD schema file.
   d. Specify the root element for the XSD schema file. For many XSD schemas for Integration Objects, the root element is SiebelMessage.
   e. Specify whether the XSD schema is for an Integration Object.
      **Important:** You must verify that this check box is selected.

4. Click **OK**.

   The Service node is listed under the Business Service object, as shown in Figure 2–34.

   **Figure 2–34 Service Node Listed Under The Business Service Object**

   You can right-click this node to create a Web service. The request and response schemas are displayed in the right pane.

   The following procedure describes how to create a Web service for a Business Object.

### 2.11 Creating and Testing a Web Service (BSE Configurations Only)

You can generate a business service (also known as a Web service) for Siebel objects you want to use with your adapter after you have properly configured the servlet BSE.

---

**Note:** In a J2EE Connector Architecture (J2CA) implementation of adapters, Web services are not available. When the adapters are deployed to use Oracle Adapter J2CA, the Common Client Interface provides integration services using the adapters.
This section contains the following topics:

- "Creating a Web Service" on page 2-25
- "Testing a Web Service" on page 2-26

2.11.1 Creating a Web Service

To generate a Web service for a Siebel Business Object:

1. Connect to your Siebel system.
2. Expand a Business Object node.
3. Expand the Business Component for which you want to create a Web service, as shown in Figure 2–35.

![Figure 2–35  Account Business Object with queryWithView method](image)  

4. Expand the object and select a method for creating the Web service, for example, queryWithView under Account.
5. Right-click the node from which you want to create a business service and select Create Business Service.

   The Create Web Service dialog is displayed.

   You can add the business object as a method for a new Web service or as a method for an existing one. Perform the following steps:

   a. From the Existing Service Names list, select either <new service> or an existing service.

   b. Specify a service name if you are creating a new service. This name identifies the Web service in the list of services under the Business Services node.

   c. Enter a description for the service (optional).

   d. Select one of the available licenses.

6. Click Next.

   The License and Method dialog is displayed. Perform the following steps:

   a. In the License field, select one or more license codes to assign to the Web service. To select more than one, hold down the Ctrl key and click the licenses.

   b. In the Method Name field, enter a descriptive name for the method.

   c. In the Description field, enter a brief description of the method.

7. Click OK.

   Application Explorer switches the view to the Business Services node, and the new Web service appears in the left pane.
2.11.2 Testing a Web Service

After you create a Web service for the Siebel Business Object, test it to ensure it functions properly. Application Explorer includes a test tool for testing a Web service.

**Testing a Web Service for a Business Object**

1. In the left pane of Application Explorer, expand the **Business Services** node.
2. Expand the **Services** node.
3. As shown in **Figure 2–36**, select the name of the business service you want to test.

**Figure 2–36  Expanded Service Node**

4. Expand the **Methods** node under the service and select the method you want to test.

The test option appears in the right pane.

If you are testing a Web service that requires XML input, then an input field is displayed.

5. **Click Invoke**.

Application Explorer displays the results in the results pane, as shown in **Figure 2–37**.
Testing a Web Service for a Business Service

After you create a Web service for the Siebel Business Service, test it to ensure it functions properly. Application Explorer includes a test tool for testing a Web service.

1. Expand the Business Services node.
2. Expand the Services node.
3. Select the name of the business service you want to test.
4. Expand the Methods node and select the name of the method you want to test.

   The test option appears in the right pane.

   If you are testing a Web service that requires XML input, then an input field is displayed.

5. Provide the appropriate input.
6. Click Invoke.

   Application Explorer displays the results in the results pane.

Identity Propagation

If you test or execute a Web service using a third party XML editor, for example XMLSPY, then the user name and password values that you specify in the SOAP header must be valid and are used to connect to Siebel. The user name and password values that you provided for Siebel during target creation using Application Explorer are overwritten for this Web service request. The following is a sample SOAP header that is included in the WSDL file for a Web service:

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<SOAP-ENV:Envelope
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Body>
    <QueryWithViewResponse
     cid="63ED62A7002CDA3D0492896446C44D0">
      <SiebelResponse status="success">
        <record>
          <Name>SIEBEL1 ACCOUNT</Name>
          <Location>ONE</Location>
        </record>
        <record>
          <Name>SIEBEL2 ACCOUNT</Name>
          <Location>TWO</Location>
        </record>
        <record>
          <Name>SIEBEL3</Name>
          <Location>RR</Location>
        </record>
      </SiebelResponse>
    </QueryWithViewResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```
<SOAP-ENV:Header>
  <m:ibsinfo xmlns:m="urn:schemas-iwaysoftware-com:iwse">
    <m:service>String</m:service>
    <m:method>String</m:method>
    <m:license>String</m:license>
    <m:disposition>String</m:disposition>
    <m:Username>String</m:Username>
    <m:Password>String</m:Password>
    <m:language>String</m:language>
  </m:ibsinfo>
</SOAP-ENV:Header>

You can remove the `<m:disposition>` and `<m:language>` tags from the SOAP header, since they are not required.

2.12 Generating WSDL (J2CA Configurations Only)

The Web Service Definition Language (WSDL) description of a Web service enables you to make the service available to other services within a host server. You use Application Explorer to create both request-response (outbound) and event notification (inbound) JCA services of the adapter.

**Note:** The Create Inbound JCA Service (Event) option is only available when the selected node supports events.

To generate a WSDL file for request-response service:

1. Under your connected Siebel target, expand Business Object, Account, Account. Navigate to an object and right-click the object.
   
The following menu is displayed, as shown in Figure 2–38.

   **Figure 2–38  Create Outbound JCA Service (Request/Response) Option**

   ![Export Schema(s)

   Create Outbound JCA Service(Request/Response)

   Apply Filter](image)

2. Select Create Outbound JCA Service (Request/Response).
   
   As shown in Figure 2–39, the Export WSDL dialog is displayed.
Figure 2–39   Export WSDL Dialog

3. Accept the default name for the file.
   The .wsdl file extension is added automatically. By default, the names of WSDL files generated for request-response services end with _invoke, while those generated for event notification end with _receive.

4. Click OK.
   The WSDL file is saved in the specified location.
   The procedure for generating WSDL for event notification is similar to request-response. To generate WSDL for event notification, you must first create a channel for every event.

2.13 Configuring an Event Adapter

Events are generated by a specific business condition being satisfied or triggered in the Siebel system. You can use events to trigger an action in your application. For example, an update to a database can reflect an update to customer information. If your application must perform when this happens, then your application is a consumer of this event.

After you create a connection to your application system, you can add events using Application Explorer. To configure an event, you must create a channel.
A channel represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by the adapter. For more information, see "Creating and Modifying a Channel" on page 2-30.

Please note that adding IO node functionality is not applicable in event configurations.

2.13.1 Creating and Modifying a Channel

The following procedure describes how to create a channel for your event. All defined event ports must be associated with a channel.

When you create, modify, or delete a channel, you must restart the Oracle WebLogic Server to recognize the change and update the repository for run-time purposes. After successfully creating the channel and inbound WSDL file, close Application Explorer before you restart the Oracle WebLogic Server.

Note: If you are planning to integrate Oracle Application Adapter for Siebel with BPEL inbound or Mediator inbound process components, then do not start the channel. The channel is managed by the run-time server after the BPEL or Mediator process component is deployed. If you start the channel from Application Explorer for testing and debugging purposes, then stop it before run-time (when working with BPEL or Mediator process components).

Three channel types are available:

- HTTP
- MQ Series
- File

Note: Channels can be configured only on the system where the Oracle Application Adapter for Siebel is installed.

2.13.1.1 Creating an HTTP Channel

To create an HTTP channel:

1. Click the Events node.
   
The Events window is displayed. The adapters that appear in the left pane support events.

2. In the left pane, expand the Siebel node.
   
The ports and channels nodes appear.

3. Right-click channels and select Add channel.
Perform the following steps:

a. Enter a name for the channel, for example, NewChannel.

b. Enter a brief description.

c. From the Protocol list, select **HTTP Listener**.

4. Click **Next**.

The Http Listener dialog is displayed, as shown in **Figure 2–41**.

5. Enter the system information as specified in the following table:
6. Click the PreParser tab, as shown in Figure 2–42.

**Figure 2–42 PreParser Tab**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listener port</td>
<td>Port on which to listen for Siebel event data.</td>
</tr>
<tr>
<td>Https</td>
<td>For a secure HTTP connection, select the Https check box.</td>
</tr>
<tr>
<td></td>
<td>This option is currently not supported.</td>
</tr>
<tr>
<td>Synchronization</td>
<td>Select REQUEST_RESPONSE from the Synchronization Type list, which is</td>
</tr>
<tr>
<td>Type</td>
<td>the recommended option.</td>
</tr>
<tr>
<td>Encoding Type</td>
<td>Choose an encoding type to be used from the Encoding Type list. By default,</td>
</tr>
<tr>
<td></td>
<td>ASCII is selected.</td>
</tr>
</tbody>
</table>

7. Specify the location of the schema file that was generated for the Integration Object node using the Export Schema(s) option in Application Explorer.

**Note:** During run time, the Oracle Application Adapter for Siebel adds the namespace to the Siebel published document using the schema that is specified in the PreParser tab. If the Schema location field in the PreParser tab is left blank, then BPEL and Mediator processes do not work properly as the Siebel published documents do not contain any namespaces.

8. Click OK.

A summary is displayed, which provides the channel description, channel status, and available ports. All the information is associated with the channel you created. The channel also appears under the channels node in the left pane, as shown in Figure 2–43.

**Figure 2–43 Inactive SiebelHTTP Node**
An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

9. Right-click the channel and select **Start**.
   The channel you created becomes active. The X over the icon in the left pane disappears.

10. To stop the channel, right-click the channel and select **Stop**.

### 2.13.1.2 Creating an MQ Series Channel

To create an MQ Series channel:

1. Click the **Events** node.
   The Events window is displayed. The adapters that appear in the left pane support events.

2. In the left pane, expand the **Siebel** node.
   The ports and channels nodes appear.

3. Right-click the **channels** node and select **Add channel**.
   The Add a new channel pane is displayed. Perform the following steps:
   a. Enter a name for the channel, for example, NewChannel.
   b. Enter a brief description.
   c. From the Protocol list, select **MQ Series Listener**.

4. Click **Next**.
   The MQ Listener dialog is displayed, as shown in **Figure 2–44**.

**Figure 2–44  MQ Listener Dialog**

![MQ Listener Dialog](image)

5. Enter the system information as specified in the following steps:
   a. In the Request tab, enter values for the following parameters:
### Parameter | Description
--- | ---
Queue manager name | The host on which the MQ Server is located (MQ Client only).
MQ server host for MQClient operation | Port on which the host database is listening.
MQ server port for MQClient operation | The number to connect to an MQ Server queue manager (MQ client only).
MQ server channel for MQClient operation | The case-sensitive name of the channel that connects with the remote MQ Server queue manager (MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.
Document type | XML
Leave the default selection.
Request queue name | Queue where the message is routed and where request documents are received. The name of the queue is case-sensitive and conforms to the following format:
Host\queue type$qName
Host
Is the system name where the MQ Series queuing system is running.
queue type
Private queues are queues that are not published in Active Directory and appear only on the local computer where they reside. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue.
qName
Is the name of the queue where messages are placed, for example,
iwaykxc1\Private\siebel

b. In the Response tab, enter values for the following parameters:

### Parameter | Definition
--- | ---
Synchronization Type | Select REQUEST_RESPONSE from the Synchronization Type list, which is the recommended option.

c. In the Advanced tab, enter values for the following parameters:

### Parameter | Definition
--- | ---
Message wait interval (msec) | The interval (in milliseconds) when to check for new input. The default is 3 seconds. Optional.
Mode of operation | Choose Sequential or Threaded.
- Sequential indicates single processing of requests.
- Threaded indicates processing of multiple requests simultaneously.
Thread limit | If you selected threaded processing, then indicate the maximum number of requests that can be processed simultaneously.
6. Click OK.
   A summary is displayed, which provides the channel description, channel status,
   and available ports. All the information is associated with the channel you created.
   The channel also appears under the channels node in the left pane.
   An X over the icon indicates that the channel is currently disconnected. You must
   start the channel to activate your event configuration.

7. Right-click the channel and select Start.
   The channel you created becomes active. The X over the icon in the left pane
   disappears.

8. To stop the channel, right-click the channel and select Stop.

2.13.1.3 Creating a File Channel
To create a File channel:

1. Click the Events node.
   The Events window is displayed. The adapters that appear in the left pane support
   events.

2. In the left pane, expand the Siebel node.
   The ports and channels nodes appear.

3. Right-click the channels node and select Add Channel.
   The Add Channel dialog is displayed. Perform the following steps:
   a. Enter a name for the channel, for example, NewChannel.
   b. Enter a brief description.
   c. From the Protocol list, select File Listener.

4. Click Next.
   The File Listener dialog is displayed, as shown in Figure 2–45.

Figure 2–45 File Listener Dialog

5. Enter the system information as specified in the following steps:
   a. In the Request tab, enter values for the following parameters:
Configuring an Event Adapter

In the Response tab, enter values for the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polling Location</td>
<td>The target file system location for the Siebel XML file.</td>
</tr>
<tr>
<td>File Mask</td>
<td>The file name to be used for the output file generated by this operation.</td>
</tr>
</tbody>
</table>

In the Advanced tab, enter values for the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synchronization Type</td>
<td>Select REQUEST_RESPONSE from the Synchronization Type list, which is the recommended option.</td>
</tr>
<tr>
<td>Response/Ack Directory</td>
<td>Directory where responses or acknowledgments are sent.</td>
</tr>
<tr>
<td>Error Directory</td>
<td>Directory to which documents with errors are written.</td>
</tr>
<tr>
<td>Poll interval (msec)</td>
<td>The interval (in milliseconds) when to check for new input. The default is 3 seconds. Optional.</td>
</tr>
<tr>
<td>Processing Mode</td>
<td>Choose Sequential or Threaded.</td>
</tr>
<tr>
<td></td>
<td>- Sequential indicates single processing of requests.</td>
</tr>
<tr>
<td></td>
<td>- Threaded indicates processing of multiple requests simultaneously.</td>
</tr>
<tr>
<td>Thread limit</td>
<td>If you selected threaded processing, then indicate the maximum number of requests that can be processed simultaneously.</td>
</tr>
</tbody>
</table>

6. Click OK.

A summary is displayed, which provides the channel description and channel status. All the information is associated with the channel you created. The channel also appears under the channels node in the left pane.

An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

7. Right-click the channel and select Start.

The channel you created becomes active.

The X over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select Stop.

2.13.1.4 Editing a Channel

To edit a channel:

1. In the left pane, select the channel you want to edit.

2. Right-click the channel and select Edit.

   The Edit channels dialog is displayed.

3. Make the required changes to the channel configuration and click OK.
2.13.1.5 Deleting a Channel
To delete a channel:

1. In the left pane, select the channel you want to delete.
2. Right-click the channel and select **Delete**.
   
The channel disappears from the list in the left pane.
Configuring an Event Adapter
This chapter describes Oracle WebLogic Server (OracleWLS) deployment and integration with Oracle Application Adapter for Siebel. It contains the following topics:

- Section 3.1, "Adapter Integration with Oracle WebLogic Server"
- Section 3.2, "Deployment of Adapter"
- Section 3.3, "Updating Adapter Configuration"

See Also:

- Oracle Application Server Adapter Concepts Guide

### 3.1 Adapter Integration with Oracle WebLogic Server

Oracle Application Adapter for Siebel is deployed within an OracleWLS container during installation. All client applications run within the OracleWLS environment. In a J2CA deployment, the Common Client Interface (CCI) integrates an OracleWLS client application with a resource adapter.

See Also:

- Oracle Application Server Adapter Concepts Guide

### 3.2 Deployment of Adapter

Figure 3–1 shows deployment of the Connector to the Oracle WebLogic Server. In a run-time service scenario, an Enterprise Java Bean, Servlet, or Java program client makes CCI calls to J2CA resource adapters. The adapters process the calls as requests and send them to the EIS. The EIS response is then sent back to the client.
3.3 Updating Adapter Configuration

During the J2CA deployment of OracleAS Adapter for Siebel, OracleWLS generates a deployment descriptor called ra.xml, located in:

<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\iwafjca.rar\META-INF

Your installation contains more than one file named ra.xml. The OracleWLS deployment descriptor that is described in this section is located in the directory specified above.

**Note:** Multiple managed connection factories are supported only for outbound processing (services).

Creating a Managed Connector Factory Object

The ra.xml descriptor provides OracleWLS-specific deployment information for resource adapters. For example, the default jca_sample configuration in Application Explorer is represented in the ra.xml file as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE connector PUBLIC "-//Sun Microsystems, Inc.//DTD Connector 1.0//EN" 'http://java.sun.com/dtd/connector_1_0.dtd'>
```
<connector>
  <display-name>IWAFJCA10</display-name>
  <vendor-name>IWAY Software</vendor-name>
  <spec-version>1.0</spec-version>
  <eis-type>IWAF</eis-type>
  <version>1.0</version>
  <license>
    <license-required>false</license-required>
  </license>
  <resourceadapter>
    <managedconnectionfactory-class>com.ibi.afjca.spi.IWAFManagedConnectionFactory</managedconnectionfactory-class>
  </resourceadapter>
  <connectionfactory-interface>javax.resource.cci.ConnectionFactory</connectionfactory-interface>
  <connectionfactory-impl-class>com.ibi.afjca.cci.IWAFConnectionFactory</connectionfactory-impl-class>
  <connection-interface>javax.resource.cci.Connection</connection-interface>
  <connection-impl-class>com.ibi.afjca.cci.IWAFConnection</connection-impl-class>
  <transaction-support>NoTransaction</transaction-support>
  <config-property>
    <config-property-name>AdapterName</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>Config</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayHome</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayConfig</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>jca_sample</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoDriver</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoURL</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoUser</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
</connector>
Updating Adapter Configuration

The parameters defined in the ra.xml file are described in the following table:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IWayHome</td>
<td>The base installation directory for the OracleWLS packaged application adapter.</td>
</tr>
<tr>
<td>IWayConfig</td>
<td>The adapter configuration name as defined in Application Explorer. For example, Oracle Application Adapter for Siebel has a preconfigured jca_sample configuration in Application Explorer.</td>
</tr>
<tr>
<td>IWayRepoURL</td>
<td>The URL to use when opening a connection to the database. This is necessary only when using an Oracle database as the repository.</td>
</tr>
<tr>
<td>IWayRepoUser</td>
<td>User name to use when connecting to the database. This is necessary only when using an Oracle database as the repository.</td>
</tr>
<tr>
<td>IWayRepoPassword</td>
<td>Password. If provided, then it overwrites configuration. This is necessary only when using an Oracle database as the repository.</td>
</tr>
<tr>
<td>loglevel</td>
<td>It overwrites the level set by the ManagedConnectorFactory property.</td>
</tr>
</tbody>
</table>

Creating Multiple Managed Connector Factory Objects

To establish multiple managed connector factory objects, you must edit the weblogic-ra.xml file and add more <connection-instance> nodes. This file is located in:

<ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/iwafjca.rar/META-INF

For example, the first jca_configuration in Application Explorer is represented in the weblogic-ra.xml file as follows:

```xml
<?xml version="1.0"?>
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">
  <enable-access-outside-app>true</enable-access-outside-app>
  <enable-global-access-to-classes>true</enable-global-access-to-classes>
  <outbound-resource-adapter>
    <default-connection-properties>
      <config-property>
        <config-property-name>IWayRepoPassword</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value></config-property-value>
      </config-property>
      <config-property>
        <config-property-name>LogLevel</config-property-name>
        <config-property-type>java.lang.String</config-property-type>
        <config-property-value>DEBUG</config-property-value>
      </config-property>
    </default-connection-properties>
    <authentication-mechanism>
      <authentication-mechanism-type>BasicPassword</authentication-mechanism-type>
      <credential-interface>javax.resource.spi.security.PasswordCredential</credential-interface>
    </authentication-mechanism>
    <reauthentication-support>true</reauthentication-support>
  </resourceadapter>
</connector>
```
To create multiple managed connector factory objects, you must add new `<connection-instance>` nodes in the file. For example:

```xml
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">
  ...
  <outbound-resource-adapter>
    ...
    <default-connection-properties>
      <pool-params>
        <initial-capacity>0</initial-capacity>
      </pool-params>
      <transaction-support>LocalTransaction</transaction-support>
    </default-connection-properties>
    ...
    <connection-instance>
      <jndi-name>eis/OracleJCAAdapter/DefaultConnection</jndi-name>
    </connection-instance>
    ...
    <connection-instance>
      <jndi-name>eis/OracleJCAAdapter/DefaultConnection1</jndi-name>
    </connection-instance>
    ...
  </outbound-resource-adapter>
</weblogic-connector>
```
If you do not specify a `<property>` element in the `<connection-instance>` section, then the value is taken from the `ra.xml` file. You can specify the default properties in the `ra.xml` file and then override them as required in the `weblogic-ra.xml` file. In addition, note that the J2CA configuration (for example, `jca_sample2`) must already be created in Application Explorer.

**Note:** When you modify the `ra.xml` and `weblogic-ra.xml` files, the Oracle WebLogic Server must be restarted. If the Oracle WebLogic Server is already running, then stop the Oracle WebLogic Server and then restart it.

In addition, the `iwafjca.rar` file must be redeployed in the Oracle WebLogic Administration Console to activate these changes.

### Modifying WSDL Files for Additional Connection Factory Values

Application Explorer generates the J2CA properties file using the default connection factory name `eis/OracleJCAAdapter/DefaultConnection`. If you created additional connection factories, then the WSDLs generated for the additional configuration and connection factory must be changed to reflect the location field of the `jca:address` section in the J2CA properties file. The default J2CA properties file for the Oracle Application Adapter for Siebel with a configuration of `isdsrv2_conn2` is shown in the following example.

Notice that the J2CA properties file has the following default connection factory:

```
eis/OracleJCAAdapter/DefaultConnection
```

```
<jca:address location="eis/OracleJCAAdapter/DefaultConnection"
            ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"
            cs.AdapterName="Siebel" cs.Config="isdsrv2_conn2"
            UIConnectionName="Connection1"/>
```

The connection factory value must be changed to the following:

```
eis/OracleJCAAdapter/DefaultConnection1
```

For example:

```
<jca:address location="eis/OracleJCAAdapter/DefaultConnection1"
            ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"
            cs.AdapterName="Siebel" cs.Config="isdsrv2_conn2"
            UIConnectionName="Connection1"/>
```

Note that only the value for the location field in the `jca:address` section should be modified. Do not modify any other field or section.
Integration With BPEL Service Components in the Oracle SOA Suite

Oracle Application Adapter for Siebel integrates seamlessly with Business Process Execution Language (BPEL) Process Manager to facilitate Web service integration. Oracle BPEL Process Manager is based on the Service-Oriented Architecture (SOA). It consumes adapter services exposed as Web Service Definition Language (WSDL) documents.

This chapter contains the following topics:

- Section 4.1, "Overview"
- Section 4.2, "Deployment of Adapter"
- Section 4.3, "Configuring a New Application Server Connection"
- Section 4.4, "Designing an Outbound BPEL Process for Service Integration"
- Section 4.5, "Designing an Inbound BPEL Process for Event Integration"

4.1 Overview

To integrate with Oracle BPEL Process Manager, Oracle Application Adapter for Siebel must be deployed in the same WLS container as Oracle BPEL Process Manager. The underlying adapter services must be exposed as WSDL files, which are generated during design time in Oracle Adapter Application Explorer (Application Explorer) for both request-response (outbound) and event notification (inbound) services of the adapter. For more information, see "Generating WSDL (J2CA Configurations Only)" on page 2-28.

The generated WSDL files are used to design the appropriate BPEL processes for inbound or outbound adapter services. A completed BPEL process must be successfully compiled in Oracle JDeveloper and deployed to a BPEL server. Upon deployment to the BPEL server, every newly built process is automatically deployed to the Oracle Enterprise Manager console, where you run, monitor, administer BPEL processes, and listen to adapter events.

4.2 Deployment of Adapter

During installation, Oracle Application Adapter for Siebel is deployed as a J2CA 1.0 resource adapter within the WLS container. The adapter must be deployed in the same WLS container as Oracle BPEL Process Manager.
4.3 Configuring a New Application Server Connection

To configure a new Application Server connection in Oracle JDeveloper:

1. Open Oracle JDeveloper on your system.
2. From the menu bar, click View and select Application Server Navigator, as shown in Figure 4–1.

![Figure 4–1 Application Server Navigator Option](image)

The Application Server tab is displayed, as shown in Figure 4–2.

![Figure 4–2 Application Server Tab](image)

3. Right-click Application Servers, and then select New Application Server.

The Create Application Server Connection Wizard - Usage page is displayed, as shown in Figure 4–3.
4. Accept the default selection (Standalone Server) and click Next. The Name and Type page is displayed, as shown in Figure 4–4.

Figure 4–4 Name and Type Page
5. Specify a new name for the Application Server connection and click Next.

The Authentication page is displayed, as shown in Figure 4–5.

Figure 4–5 Authentication Page

6. Specify a valid user name (for example, weblogic) and a password (for example, welcome1) for your new connection.

7. Click Next.

The Configuration page is displayed, as shown in Figure 4–6.
Specify the Oracle WebLogic host name (for example, localhost), which is the machine IP where the process needs to deploy and Oracle WebLogic domain (for example, base_domain).

Click Next.

The Test page is displayed, as shown in Figure 4–7.
10. Click **Test Connection**.

11. Ensure that the test status is successful.

12. Click **Next**.

   The Finish page is displayed, as shown in **Figure 4–8**.
13. Click Finish.

As shown in Figure 4–9, the new Application Server connection is listed in the left pane (Application Server tab).

*Figure 4–9  The New Application Server Connection*

---

### 4.4 Designing an Outbound BPEL Process for Service Integration

The following tools are required to complete your adapter design-time configuration:

- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPEL Designer (JDeveloper) or Eclipse

**Note:** The examples in this chapter demonstrate the use of Oracle JDeveloper.
Before you design a BPEL process, you must create a schema and generate the respective WSDL file using Application Explorer. For more information, see "Generating WSDL (J2CA Configurations Only)" on page 2-28.

### 4.4.1 Siebel Service Integration

This section describes how to perform service integration with Siebel. The following topics are included:

- Creating a Request-Response J2CA Service in Application Explorer
- Creating an Integration Object Node

#### Creating a Request-Response J2CA Service in Application Explorer

To generate WSDL in Application Explorer:

1. Start Application Explorer and connect to a defined Siebel target or create a new target.
   
   For more information, see Connecting to a Defined Target on page 2-10.

2. Expand the Siebel target to which you are connected.

3. As shown in Figure 4–10, expand Business Object, Account, and then Account.

#### Figure 4–10 Create Outbound JCA Service (Request/Response) Option

4. Right-click queryWithView, and then select Create Outbound JCA Service (Request/Response).
   
   The Export WSDL dialog is displayed, as shown in Figure 4–11.
Figure 4–11 Export WSDL Dialog

5. Specify an export location on your file system or accept the default path. The .wsdl file extension is added automatically. By default, the names of WSDL files generated for request-response services end with _invoke.

6. Click OK.

You can now create an empty composite for SOA, which is the first step that is required to define a BPEL outbound process in Oracle JDeveloper.

4.4.2 Creating a New SOA Application for the Outbound BPEL Process

Perform the following steps to create a new SOA application for the outbound BPEL process:

1. Open Oracle JDeveloper on your system.

2. Click Application in the menu bar and select New from the menu, as shown in Figure 4–12.
Figure 4–12  New Option in Application Menu

The Create SOA Application wizard is displayed, as shown in Figure 4–13.
3. From the Application Template list, click **SOA Application**.

4. Enter a name for the new SOA application (for example, Siebel_Outbound_BPEL) and click **Next**.

   The Name your project page is displayed, as shown in Figure 4–14.
5. Enter a project name (for example, QueryWithView_Invoke) and click Next.

The Configure SOA settings page is displayed, as shown in Figure 4–15.
6. From the Composite Template list, select **Empty Composite** and click **Finish**.

As shown in Figure 4–16, the new SOA application (Siebel_Outbound_BPEL) and associated project (QueryWithView_Invoke) are added to the Application Navigator tab in the left pane.

**Figure 4–16  SOA application (Siebel_Outbound_BPEL) In Application Navigator Tab**
4.4.3 Defining a BPEL Outbound Process

This section describes how to define a BPEL outbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Outbound BPEL Process Component

Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the Third Party Adapter component from the Component Palette tab (Service Adapters section) to the External References design pane, as shown in Figure 4–17.

Figure 4–17 Third Party Adapter Component

The Create Third Party Adapter Service dialog is displayed, as shown in Figure 4–18.
2. Enter a name for the new third party adapter service.

3. Ensure that **Reference** is selected from the Type list (default).

4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed, as shown in Figure 4-19.
5. Browse and select an outbound WSDL file from the following directory:
   <ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/wsdls

6. Click OK.
   The Localize Files dialog is displayed, as shown in Figure 4–20.
7. Click OK.

The outbound WSDL file and associated request and response XML schema files (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 4–21.

Figure 4–21 Create Third Party Adapter Service Dialog
8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in Figure 4–22.

*Figure 4–22  SOA Resource Browser Dialog*

9. Browse and select the JCA properties file from the following directory:

```
<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls
```

10. Click **OK**.

As shown in Figure 4–23, the following message is displayed.

*Figure 4–23  Copy File Confirmation Message*

```
The jca file queryWithView_invoke.jca is external to the current project. In order to make this file available to your project at runtime, JDeveloper can now make a local copy. Do you want to copy file to queryWithView_invoke_3P.jca?

Yes  No
```
11. Click Yes.

A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 4–24.

**Figure 4–24 Create Third Party Adapter Service Dialog**

![Create Third Party Adapter Service Dialog](image)

12. Click OK.

The third party adapter service component (GetDetail) is created and displayed in the External References pane, as shown in Figure 4–25.
You are now ready to configure an outbound BPEL process component.

**Configuring an Outbound BPEL Process Component**

Perform the following steps to configure an outbound BPEL process component:

1. Drag and drop the `BPEL Process` component from the Component Palette tab (Service Components section) to the Components design pane, as shown in Figure 4–26.

The Create BPEL Process dialog is displayed, as shown in Figure 4–27.
2. In the Name field, enter a name to identify the new outbound BPEL process component (for example, querywithview_out).

3. From the Template list, select Synchronous BPEL Process.

4. Click the Browse Input Elements icon, which is located to the right of the Input field to select the associated XML request schema file.

   The Type Chooser dialog is displayed, as shown in Figure 4–28.

---

**Figure 4–27  Create BPEL Process Dialog**

**Figure 4–28  Type Chooser Dialog**
5. Expand Project Schema Files, `queryWithView_invoke_request.xsd`, and then select **Siebel**.

6. Click OK.

   You are returned to the Create BPEL Process dialog, as shown in Figure 4–29.

**Figure 4–29  Create BPEL Process Dialog**

7. Click the **Browse Output Elements** icon, which is located to the right of the Output field to select the associated XML response schema file.

   The Type Chooser dialog is displayed, as shown in Figure 4–30.

**Figure 4–30  Type Chooser Dialog**
8. Expand Project Schema Files, queryWithView_invoke_response.xsd, and then select SiebelResponse.

9. Click OK.

You are returned to the Create BPEL Process dialog, as shown in Figure 4–31.

**Figure 4–31 Create BPEL Process Dialog**

10. Click OK.

11. As shown in Figure 4–32, create a connection between the outbound BPEL process component (querywithview_out) and the third party adapter service component (QueryWithView).

**Figure 4–32 Connection Between the Outbound BPEL Process Component (querywithview_out) and the Third Party Adapter Service Component (QueryWithView)**
12. Double-click the outbound BPEL process component (querywithview_out) in the Components pane.

The following is displayed, as shown in Figure 4–33.

**Figure 4–33  Outbound BPEL Process Component (querywithview_out) in the Components Pane**

13. Drag and drop the **Invoke** activity component to the Components pane and place it between the **receiveInput** activity component and the **replyOutput** activity component, as shown in Figure 4–34.
14. Create a connection between the new Invoke activity component (Invoke_1) and the third party adapter service component (QueryWithView), as shown in Figure 4–35.

The Edit Invoke dialog is displayed, as shown in Figure 4–36.
15. Click the **Automatically Create Input Variable** icon, which is located to the right of the Input field to configure a new input variable.

The Create Variable dialog is displayed, as shown in Figure 4–37.

**Figure 4–37  Create Variable Dialog**

16. Accept the default values that are provided for the new input variable and click **OK**.

You are returned to the Edit Invoke dialog, as shown in Figure 4–38.
17. Click the **Automatically Create Output Variable** icon, which is located to the right of the Output field to configure a new output variable.

   The Create Variable dialog is displayed, as shown in **Figure 4–39**.

**Figure 4–39  Create Variable Dialog**

![Create Variable Dialog](image)

18. Accept the default values that are provided for the new output variable and click **OK**.

   You are returned to the Edit Invoke dialog, as shown in **Figure 4–40**.
19. Click **Apply** and then **OK**.

   As shown in Figure 4–41, the Invoke activity component (Invoke_1) is updated accordingly.
20. Drag and drop the Assign activity component to the Components pane and place it between the Receive activity component (receiveInput) and the Invoke activity component (Invoke_1), as shown in Figure 4–42.

21. Double-click the new Assign activity component (Assign_1).

The Assign dialog is displayed, as shown in Figure 4–43.
22. Click the **Copy Operation** tab.

23. Click the **Plus sign** icon and select **Copy Operation** from the list of available operations.

    The Create Copy Operation dialog is displayed, as shown in Figure 4–44.

**Figure 4–43 Assign Dialog**

**Figure 4–44 Create Copy Operation Dialog**
24. In the From pane, expand Variables, InputVariable, and then select payload.

25. In the To pane, expand Variables, Invoke_1_queryWithView_InputVariable, and then select input_queryWithView.

26. Click OK.

You are returned to the Assign dialog, as shown in Figure 4–45.

**Figure 4–45 Assign Dialog**

27. Click Apply and then OK.

28. As shown in Figure 4–46, drag and drop the Assign activity component to the Components pane and place it between the Invoke activity (Invoke_1) and the Reply activity (replyOutput).
29. Double-click the new Assign activity component (Assign_2). The Assign dialog is displayed, as shown in Figure 4–47.

Figure 4–46 Assign Activity Component Dragged to the Components Pane and Placed Between the Invoke Activity (Invoke_1) and the Reply Activity (replyOutput)

30. Click the Copy Operation tab.

31. Click the Plus sign icon and select Copy Operation from the list of available operations. The Create Copy Operation dialog is displayed, as shown in Figure 4–48.
32. In the From pane, expand Variables, Invoke_1_queryWithView_OutputVariable, and then select output_queryWithView.

33. In the To pane, expand Variables, outputVariable, and then select payload.

34. Click OK.

You are returned to the Assign dialog, as shown in Figure 4–49.
Figure 4–49  Assign Dialog

35. Click **Apply** and then **OK**.

The completed activity flow is now displayed, as shown in Figure 4–50.
36. As shown in Figure 4–51, double-click composite.xml in the left pane.

37. As shown in Figure 4–52, click the Save All icon in the menu bar to save the new outbound BPEL process component that was configured.
You are now ready to deploy the BPEL outbound process.

4.4.4 Deploying the BPEL Outbound Process

Perform the following steps to deploy the BPEL outbound process.

1. Right-click the project name in the left pane (for example, QueryWithView_Invoke), select Deploy, and then click QueryWithView_Invoke, as shown in Figure 4–53.

The Deployment Action dialog is displayed, as shown in Figure 4–54.
2. Ensure that **Deploy to Application Server** is selected.

3. Click **Next**.

   The Deploy Configuration dialog is displayed, as shown in **Figure 4–55**.
4. Leave the default values selected and click **Next**.

   The Select Server dialog is displayed, as shown in **Figure 4–56**.

**Figure 4–56 Select Server Dialog**
5. Select an available application server that was configured and click Next.

The SOA Servers dialog is displayed, as shown in Figure 4–57.

**Figure 4–57 SOA Servers Dialog**

6. Select a target SOA server and click Next.

The Summary dialog is displayed, as shown in Figure 4–58.
7. Review and verify all the available deployment information for your project and click **Finish**.

The process is deployed successfully, as shown in Figure 4–59.

**Figure 4–59  Successful Deployment Message**

```
Validating composite : C:\Developer\mywork\Siebel_Outbound_BPEL\QueryWVIl
BUILD SUCCESSFUL
Total time: 16 seconds
```

### 4.4.5 Invoking the Input XML Document in the Oracle Enterprise Manager Console

Perform the following steps to invoke the input XML document in the Oracle Enterprise Manager console.

1. Log in to the Oracle Enterprise Manager console by using the following URL:
   ```
   http://localhost:7001/em
   ```

2. Expand your domain in the left pane followed by the **SOA** folder, as shown in Figure 4–60.
3. As shown in Figure 4-61, select an available project (for example, Siebel_Outbound_queryWithView_BPEL).

4. Click Test in the right pane.
   The Test Web Service page is displayed, as shown in Figure 4-62.
5. Click the Request tab.
6. As shown in Figure 4–63, scroll down to the Input Arguments section.

7. Select XML View from the list in the upper-left corner.
9. Click Test Web Service.

The output response is received in the Response tab of the Oracle Enterprise Manager console, as shown in Figure 4–64.
Designing an Inbound BPEL Process for Event Integration

4.4.6 Testing Outbound BPEL and Mediator Processes

When testing an outbound BPEL process from the Oracle Enterprise Manager console or an outbound Mediator process from the Enterprise Manager console, do not use the XML envelopes that are generated by these consoles. Instead, remove them and use the XML payloads that are generated from the schemas, which conform to the WSDLs for namespace qualifications.

The Mediator data flows can be tested using the Enterprise Manager console. When creating a Mediator data flow and interactions, the Web services are created and registered with the Oracle Application Server. For more information on creating a Mediator outbound process, see Chapter 5, “Integration With Mediator Service Components in the Oracle SOA Suite”.

4.5 Designing an Inbound BPEL Process for Event Integration

This topic illustrates Siebel event integration. It describes design-time and run-time configuration. The following tools are required to complete your adapter design-time configuration:

- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPEL Designer (JDeveloper) or Eclipse

**Note:** The examples in this chapter demonstrate the use of Oracle JDeveloper.

Before you design a BPEL process, you must generate the respective WSDL file using Application Explorer. For more information, see “Generating WSDL for Event Integration” on page 4-44.
4.5.1 Generating WSDL for Event Integration

You must create a separate channel for every inbound J2CA service and select that channel when you generate WSDL for inbound interaction using Application Explorer.

---

**Note:** If two or more events share the same channel, then event messages may not be delivered to the right BPEL process.

---

Creating a Channel

You must create a separate channel for every inbound J2CA service and select that channel when you generate WSDL for inbound interaction using Application Explorer.

---

**Note:** If two or more events share the same channel, then event messages may not be delivered to the right BPEL process.

---

To create a channel:

1. In the left pane, click **Events**.
2. Expand the **Siebel** node.
   
   The ports and channels nodes appear in the left pane, as shown in Figure 4–65.

**Figure 4–65  Ports and Channels Nodes Under Siebel Node**

   ![Channels and Ports Nodes](image)

3. Right-click **Channels** and select **Add Channel**.
   
   The Add Channel dialog is displayed, as shown in Figure 4–66.
Perform the following steps:

a. Enter a name for the channel, for example, SiebelEvent.

b. Enter a brief description.

c. From the Protocol list, select HTTP Listener, MQ Series Listener, or File Listener.

For demonstration purposes, this procedure uses the HTTP Listener as an example.

4. Click Next.

The Basic dialog is displayed, as shown in Figure 4–67.
5. Enter the system information as specified in the following table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listener port</td>
<td>Port on which to listen for Siebel event data.</td>
</tr>
<tr>
<td>Https</td>
<td>For a secure HTTP connection, select the Https check box.</td>
</tr>
<tr>
<td>Synchronization Type</td>
<td>Select REQUEST_RESPONSE from the list, which is the recommended option.</td>
</tr>
<tr>
<td>Encoding Type</td>
<td>Choose an encoding type to be used from the list. By default, ASCII is selected.</td>
</tr>
</tbody>
</table>

6. Click the PreParser tab, as shown in Figure 4–68.

7. Specify the location of the schema file that was generated for the Integration Object node using the Export Schema(s) option in Application Explorer.
8. Click OK.

As shown in Figure 4–69, the channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected.

**Figure 4–69  New Channel Node**

---

**Note:** During run time, the Oracle Application Adapter for Siebel adds the namespace to the Siebel published document using the schema that is specified in the PreParser tab. If the Schema location field in the PreParser tab is empty, then BPEL and Mediator processes do not work properly as the Siebel published documents do not contain any namespaces.

---

You must now create an Integration Object node.

**Creating an Integration Object Node**

1. Start Application Explorer.
2. Expand the **Adapters** node, as shown in Figure 4–70.

**Figure 4–70  Disconnected Siebel Target Node, Siebel, Under the Siebel Node**

---

Perform the following steps:

a. Expand the **Siebel** node.

The defined Siebel targets are displayed under the adapter node.

b. Click the target name, for example, siebel, under the **Siebel** node.

The Connection dialog displays the values you entered.

3. Verify your connection parameters.
4. Right-click the target name and select **Connect**.

The x icon disappears, indicating that the node is connected, as shown in Figure 4–71.
5. Expand the **Integration Object** node and select **Sample Account**.

6. Right-click the **Sample Account** node and select **Add IO Node**. The Add IO Node dialog is displayed, as shown in **Figure 4–72**.

7. Enter a node name (for example, Sample_Account) in the **Node name** field and a path to the Sample Account XSD file in the **Schema location** field.

Please note:

- **For Siebel 7.5 or later:** Generate XSD schemas directly from Siebel tools. You use the XSD schemas when you create Web services in Application Explorer. After you generate an XSD schema through Siebel tools, use it to create an IO node and a Web service.

- **For Siebel 7.0:** You cannot generate XSD schemas directly from Siebel tools; only XDR schemas can be created. Before you create a Web service, you must first generate an XSD schema from the XDR schema using Application Explorer.

8. If the XSD schema has already been generated, then select XSD Schema. If you are using Siebel-generated XDR schemas, then do not select the XSD schema option.

9. Select a protocol (HTTP, FILE, or MQ Series) from the **Protocol** list.

10. Click **Continue**.

    The new Integration Object node is added, as shown in **Figure 4–73**.
Generating WSDL for Event Notification

After you create a channel and an associated Integration Object node, you must generate WSDL for the event using Application Explorer.

You must be connected to a Siebel target under the Adapters node in Application Explorer. For detailed information on how to define and connect to a target, see "Establishing a Connection (Target) for Siebel" on page 2-6.

After you connect to a Siebel target, generate WSDL for the event as follows:

1. Right-click the Integration Object node (for example, Sample_Account), and then select Create Inbound JCA Service (Event), as shown in Figure 4–74.

![Figure 4–74 Create Inbound JCA Service (Event) Option Selected in Application Explorer](image)

The Export WSDL dialog is displayed, as shown in Figure 4–75.

---

**Note:** You must restart the Oracle WebLogic Server after the Integration Object node and channel are created.
Perform the following steps:

a. **In the Name field**, specify a name for the WSDL file.
   
   The .wsdl file extension is added automatically. By default, the names of WSDL files generated for events end with _receive.

b. **From the Channel list**, select the channel you created for this inbound service (for example, SiebelEvent).

   **Important:** You must create a separate channel for every inbound service. Verify that the channel is stopped before run-time.

2. **Click OK.**

### 4.5.2 Creating a New SOA Application for the Inbound BPEL Process

Perform the following steps to create a new SOA application for the inbound BPEL process:

1. **Open Oracle JDeveloper on your system.**
2. Click **Application** in the menu bar and select **New** from the menu, as shown in Figure 4–76.

**Figure 4–76  New Option Being Selected in the Application Menu**

The Create SOA Application wizard - Name your application page is displayed, as shown in Figure 4–77.
3. From the Application Template list, click **SOA Application**.

4. Enter a name for the new SOA application (for example, Siebel_Inbound_BPEL) and click **Next**.

   The Name your project page is displayed, as shown in **Figure 4-78**.
5. Enter a project name (for example, SampleAccount_Receive) and click Next. The Configure SOA settings page is displayed, as shown in Figure 4–79.
6. From the Composite Template list, select **Empty Composite** and click **Finish**.

As shown in Figure 4–80, the new SOA application (Siebel_Inbound_BPEL) and associated project (SampleAccount_Receive) are added to the Application Navigator tab in the left pane.

*Figure 4–80  New SOA Application (Siebel_Inbound_BPEL) and Associated Project (SampleAccount_Receive)*

**4.5.3 Defining a BPEL Inbound Process**

This section describes how to define a BPEL inbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Inbound BPEL Process Component

**Creating a Third Party Adapter Service Component**

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the Exposed Services pane, as shown in Figure 4–81.
The Create Third Party Adapter Service dialog is displayed, as shown in Example 4–82.

2. Enter a name for the third party adapter service.
3. Ensure that Service is selected from the Type list (default).
4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed, as shown in Example 4–83.
5. Browse and select an inbound WSDL file from the following directory:

   `<ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/wsdl`

6. Click OK.

   The Localize Files dialog is displayed, as shown in Example 4–84.

7. Click OK.
The inbound WSDL file and associated receive/request XML schema file (.xsd) are imported to the project folder that has been created.

As shown in Example 4–85, you are returned to the Create Third Party Adapter Service dialog.

**Figure 4–85  Create Third Party Adapter Service Dialog**

8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in Example 4–86.
9. Browse and select the JCA properties file from the following directory:

\<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls

10. Click OK.

As shown in Example 4–87, the following message is displayed.

11. Click Yes.

A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog, as shown in Example 4–88.
12. Click OK.

The third party adapter service component (SampleAccount) is created and displayed in the Exposed Services pane, as shown in Example 4–89.

You are now ready to configure an inbound BPEL process component.
Creating an Inbound BPEL Process Component

Perform the following steps to create an inbound BPEL process component:

1. Drag and drop the BPEL Process component from the Component Palette tab (Service Components section) to the Components pane, as shown in Example 4–90.

2. In the Name field, enter a name to identify the new inbound BPEL process component (for example, SampleAccount_Inbound).

3. From the Template list, select Base on a WSDL.

4. Uncheck the Expose as a SOAP service check box.

5. Click the Find existing WSDLs icon, which is located to the right of the WSDL URL field.
The SOA Resource Browser dialog is displayed, as shown in Example 4-92.

Figure 4–92  SOA Resource Browser Dialog

6. Browse and select an inbound WSDL file from the project folder (for example, Sample_Account_receive.wsdl).

7. Click OK.

The Localize Files dialog is displayed, as shown in Example 4–93.

Figure 4–93  Localize Files Dialog
8. Click OK.

You are returned to the Create BPEL Process dialog, as shown in Example 4–94.

Figure 4–94  Create BPEL Process Dialog

9. Click OK to create the connection, as shown in Example 4–95.

Figure 4–95  Connection Created Between the Third Party Adapter Service Component (SampleAccount) and the Inbound BPEL Process Component (SampleAccount_Inbound)

10. Create a connection between the third party adapter service component (SampleAccount) and the inbound BPEL process component (SampleAccount_Inbound).

11. Double-click composite.xml in the left pane, as shown in Example 4–96.
12. Click the Save All icon in the menu bar to save the new inbound BPEL process component that was configured, as shown in Example 4–97.

You are now ready to deploy the BPEL inbound process.

4.5.4 Deploying the BPEL Inbound Process

Perform the following steps to deploy the BPEL inbound process.
1. Right-click the project name in the left pane (for example, **SampleAccount_Receive**), select **Deploy** and then click **SampleAccount_Receive**, as shown in Example 4-98.

**Figure 4–98 SampleAccount_Receive Option**

![SampleAccount_Receive Option](image1.png)

The Deployment Action dialog is displayed, as shown in Example 4–99.

**Figure 4–99 Deployment Action Dialog**

![Deployment Action Dialog](image2.png)

2. Ensure that **Deploy to Application Server** is selected.
3. Click **Next**.

The Deploy Configuration dialog is displayed, as shown in Example 4–100.
4. Leave the default values selected and click Next.
   
The Select Server dialog is displayed, as shown in Example 4–101.

**Figure 4–101  Select Server Dialog**
5. Select an available application server that was configured and click **Next**.

   The SOA Servers dialog is displayed, as shown in **Example 4–102**.

**Figure 4–102  SOA Servers Dialog**

6. Select a target SOA server and click **Next**.

   The Summary dialog is displayed, as shown in **Example 4–103**.
7. Review and verify all the available deployment information for your project and click Finish.

The process is deployed successfully, as shown in Example 4–104.

4.5.5 Triggering an Event in Siebel

The following topic describes how to trigger an event in Siebel and verify event integration using Oracle Application Adapter for Siebel.

Triggering a Siebel Event to Test Event Runtime Integration

To trigger an event in Siebel:

1. As shown in Example 4–105, start the Siebel Call Center by entering the following URL in a browser:

   http://host name/callcenter/start.swe
2. Click View and select Site Map from the list.
   
   The Site Map view is displayed, as shown in Example 4–106.

3. Click Siebel Workflow Administration.
   
   The Siebel Workflow Administration page is displayed, as shown in Example 4–107.

4. Click Workflow Processes.
   
   The Workflow Processes page is displayed, as shown in Example 4–108.
5. Click **Query** to search for the Workflow needed to trigger a Siebel event, as shown in Example 4–109.
6. As shown in Example 4–110, enter a Siebel workflow name and click **Search**.

![Workflow Processes Page](image)

**Figure 4–110  Workflow Processes Page**

7. As shown in Example 4–111, select the workflow.

![Process Designer Tab](image)

**Figure 4–111  Process Designer Tab**

8. Click the **Process Designer** tab and double-click the **Send Siebel Quote Data HTTP** workflow element.

   The **Input Arguments tab** is displayed, as shown in Example 4–112.

![Input Arguments Tab](image)

**Figure 4–112  Input Arguments Tab**

9. Enter the IP address and port for the **HTTPRequestURLTemplate** input argument.

10. Click **Return To Designer**, as shown in Example 4–113.
11. Click the Process Simulator tab, as shown in Example 4–114.

![Figure 4–114  Process Simulator Tab](image)

The Simulator tab is displayed, as shown in Example 4–115.

![Figure 4–115  Simulator Tab](image)

12. Click Start then Continue to complete the Siebel event triggering process.

**Verifying the Results**
To verify your results:

1. Log in to the Oracle Enterprise Manager console by using the following URL:
http://localhost:7001/em

2. As shown in Example 4–116, expand your domain in the left pane followed by the SOA folder.

Figure 4–116 Oracle Enterprise Manager Console

3. As shown in Example 4–117, select an available project (for example, Siebel_Inbound_sampacc_ionode_BPEL).

Figure 4–117 Instances Tab

4. Click the Instances tab in the right pane.
Recently received run-time events are displayed in the Instances tab, as shown in Example 4–118.

**Figure 4–118 Recently Received Run-time Events Displayed in the Instances Tab**

<table>
<thead>
<tr>
<th>Instance ID</th>
<th>Name</th>
<th>Conversation ID</th>
<th>State</th>
<th>Start Time</th>
<th>Logs</th>
</tr>
</thead>
<tbody>
<tr>
<td>65002</td>
<td></td>
<td>1Way.126640430730</td>
<td>Completed</td>
<td>Feb 17, 2010 5:50:27 AM</td>
<td></td>
</tr>
</tbody>
</table>

5. Select a Siebel instance ID.

The Flow Trace page is displayed, as shown in Example 4–119.

**Figure 4–119 Flow Trace Page**

6. Select a component instance to view its detailed audit trail.

The Instance page for the selected component is displayed, as shown in Example 4–120.
Click the **Audit Trail** tab to view the event message.

The message received from the Siebel system is displayed in the Audit Trail tab.
Integration With Mediator Service Components in the Oracle SOA Suite

This chapter describes integration with Mediator service components in the Oracle SOA Suite. It contains the following topics:

- Section 5.1, "Configuring a New Application Server Connection"
- Section 5.2, "Configuring a Mediator Outbound Process"
- Section 5.3, "Configuring a Mediator Inbound Process"

The scenarios shown in this chapter require the following prerequisites.

**Prerequisites**
The following are installation and configuration requirements:

- Oracle Application Adapter for Siebel must be installed on Oracle WebLogic Server.
- Siebel must be configured for inbound and outbound processing.
- OracleAS Technology adapters must be deployed and properly configured.

The examples in this chapter present the configuration steps necessary for demonstrating service and event integration with Siebel. Prior to using this material, you must be familiar with the following:

- How to configure Oracle Application Adapter for Siebel for services and events. For more information, see Chapter 2, "Configuring Oracle Application Server Adapter for Siebel".
- How to configure Oracle JDeveloper. For more information, see Chapter 4, "Integration With BPEL Service Components in the Oracle SOA Suite".
- How to use Siebel workflows. For more information on Siebel design requirements, see Appendix A, "Using Siebel Workflows".

**Overview of Mediator Integration**
Mediator provides a comprehensive application integration framework. Oracle Application Adapter for Siebel used with Mediator enables you to seamlessly integrate enterprise software, eliminating the need to write custom code. Functional modeling, as opposed to custom coding solutions, allows for software reuse and reduces the complexity and management challenges that arise over the software lifecycle. This integration model consists of two components—high-level integration logic and low-level platform services.

Adapter integration with Oracle WebLogic Server, Mediator is a two-step process:
1. **Design Time**: Oracle Application Adapter for Siebel is configured in Application Explorer for services and events, as described in Chapter 2, "Configuring Oracle Application Server Adapter for Siebel". Integration logic is modeled in iStudio. Metadata are stored in repositories.

2. **Runtime**: The underlying platform treats this metadata as run-time instructions to enable the communication between participating applications.

### 5.1 Configuring a New Application Server Connection

For more information on how to configure a new Application Server connection in Oracle JDeveloper, see Section 4.3, "Configuring a New Application Server Connection".

### 5.2 Configuring a Mediator Outbound Process

The following example describes how to configure a Mediator outbound process to your Siebel system, using a Mediator project in Oracle JDeveloper.

**Prerequisites**

Before you design a Mediator outbound process, you must generate the respective WSDL file using Application Explorer. For more information, see "Siebel Service Integration" on page 4-8.

### 5.2.1 Creating a New SOA Application for the Outbound Mediator Process

Perform the following steps to create a new SOA application for the outbound Mediator process:

1. Open Oracle JDeveloper on your system.

2. Click **Application** in the menu bar and select **New** from the menu, as shown in Figure 5–1.
**Figure 5–1  New Option in the Application Menu**

![Application Menu Screenshot]

The Create SOA Application wizard - Name your application page is displayed, as shown in Figure 5–2.
3. From the Application Template list, click **SOA Application**.

4. Enter a name for the new SOA application (for example, Siebel_Outbound_Mediator) and click **Next**.
   
The Name your project page is displayed, as shown in **Figure 5–3**.
Figure 5–3 Name Your Project Page

5. Enter a project name (for example, QueryWithView_Invoke) and click Next.
   The Configure SOA settings page is displayed, as shown in Figure 5–4.

Figure 5–4 Configure SOA Settings Page
6. From the Composite Template list, select **Empty Composite** and click **Finish**.

As shown in Figure 5–5, the new SOA application (Siebel_Outbound_Mediator) and associated project (QueryWithView_Invoke) are added to the Application Navigator tab in the left pane.

*Figure 5–5  A New SOA Application (Siebel_Outbound_Mediator) and Associated Project (QueryWithView_Invoke) Added to the Application Navigator Tab*

---

### 5.2.2 Defining a Mediator Outbound Process

This section describes how to define a Mediator outbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Outbound Mediator Process Component
3. Configuring the Routing Rules

**Configuring a Third Party Adapter Service Component**

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the External References pane, as shown in Figure 5–6.
Configuring a Mediator Outbound Process

Integration With Mediator Service Components in the Oracle SOA Suite

Figure 5–6 Third Party Adapter Component Being Dragged to the External References Pane

The Create Third Party Adapter Service dialog is displayed, as shown in Figure 5–7.

Figure 5–7 Create Third Party Adapter Service Dialog

2. Enter a name for the new third party adapter service.
3. Ensure that Reference is selected from the Type list (default).
4. Click the Find existing WSDLs icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed, as shown in Figure 5–8.
5. Browse and select an outbound WSDL file from the following directory:

<ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/wsdls

6. Click OK.

The Localize Files dialog is displayed, as shown in Figure 5–9.
7. Click OK.

The outbound WSDL file and associated request and response XML schema files (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 5–10.

Figure 5–10  Create Third Party Adapter Service Dialog
8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in *Figure 5–11*.

*Figure 5–11  SOA Resource Browser Dialog*

![SOA Resource Browser Dialog](image)

9. Browse and select the JCA properties file from the following directory:

   `<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls`

10. Click **OK**.

    The following message is displayed, as shown in *Figure 5–12*.

*Figure 5–12  Copy File Message*

![Copy File Message](image)
11. Click Yes.

   A copy of the JCA properties file is made in the project folder.
   You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 5–13.

**Figure 5–13  Create Third Party Adapter Service Dialog**

![Create Third Party Adapter Service Dialog](image)

12. Click OK.

   The third party adapter service component (GetDetail) is created and displayed in the External References pane, as shown in Figure 5–14.
Configuring an Outbound Mediator Process Component

Perform the following steps to configure an outbound Mediator process component:

1. Drag and drop the **Mediator** component from the Component Palette tab (Service Components section) to the Components pane, as shown in **Figure 5–15**.

You are now ready to configure an outbound Mediator process component.

The Create Mediator dialog is displayed, as shown in **Figure 5–16**.
2. In the Name field, enter a name to identify the new outbound Mediator process component (for example, QueryWithView_Mediator).

3. From the Template list, select **Synchronous Interface**.

4. Click the **Browse Input Elements** icon, which is located to the right of the Input field to select the associated XML request schema file.

   The Type Chooser dialog is displayed, as shown in Figure 5–17.
5. Expand **Project WSDL Files**, **queryWithView_invoke.wsdl**, **Inline Schemas**, **schema**, and select **Siebel**.

6. Click **OK**.

   You are returned to the Create Mediator dialog, as shown in Figure 5–18.

**Figure 5–17  Type Chooser Dialog**

**Figure 5–18  Create Mediator Dialog**
7. Click the **Browse Output Elements** icon, which is located to the right of the Output field to select the associated XML response schema file.

The Type Chooser dialog is displayed, as shown in Figure 5–19.

**Figure 5–19  Type Chooser Dialog**

```
Type Chooser

Type: :ware:adapter:siebel:response5/BO/Account/Account/queryWithView\SiebelResponse

Show Detailed Node Information

OK  Cancel
```


9. Click **OK**.

You are returned to the Create Mediator dialog, as shown in Figure 5–20.
10. Click OK.

11. As shown in Figure 5–21, create a connection between the outbound Mediator process component (QueryWithView_Mediator) and the third party adapter service component (QueryWithView).

You are now ready to configure the routing rules.

Configuring the Routing Rules
Perform the following steps to configure routing rules for the Mediator outbound process component:
1. Double-click the outbound Mediator process component (QueryWithView_Mediator) in the Components pane, as shown in Figure 5–22.

Figure 5–22 The Outbound Mediator Process Component (QueryWithView_Mediator) in the Components Pane

The Routing Rules dialog is displayed, as shown in Figure 5–23.

Figure 5–23 Routing Rules Dialog

2. In the <<Filter Expression>> area, click the Select an existing mapper file or create a new one icon to the right of the Transform Using field.

The Request Transformation Map dialog is displayed, as shown in Figure 5–24.
3. Select the Create New Mapper File option and click OK.

   The Siebel_To_Siebel.xsl tab is displayed, as shown in Figure 5–25.


   The Auto Map Preferences dialog is displayed, as shown in Figure 5–26.
5. Retain the default values and click **OK**.

6. As shown in Figure 5–27, click the **QueryWithView_Mediator.mplan** tab.

![Figure 5–27 QueryWithView_Mediator.mplan Tab](image)

You are returned to the Routing Rules dialog, as shown in Figure 5–28.

![Figure 5–28 Routing Rules Dialog](image)

7. In the Synchronous Reply area, click the **Select an existing mapper file or create a new one** icon to the right of the Transform Using field.

   The Reply Transformation Map dialog is displayed, as shown in Figure 5–29.

![Figure 5–29 Reply Transformation Map Dialog](image)

8. Select the **Create New Mapper File** option and click **OK**.

   The **SiebelResponse_To_SiebelResponse.xsl** tab is displayed, as shown in Figure 5–30.
9. Map the `iWayResponse:SiebelResponse` source element to the `iWayResponse:SiebelResponse` target element. The Auto Map Preferences dialog is displayed, as shown in Figure 5–31.

10. Retain the default values and click **OK**. As shown in Figure 5–32, the mapping is completed.

11. Click the **Save All** icon in the menu bar to save the new outbound Mediator process component that was configured, as shown in Figure 5–33.
5.2.3 Deploying the Mediator Outbound Process

Perform the following steps to deploy the Mediator outbound process.

1. As shown in Figure 5–34, right-click the project name in the left pane (for example, QueryWithView_Invoke), select Deploy, and then click QueryWithView_Invoke.

The Deployment Action dialog is displayed, as shown in Figure 5–35.
2. Ensure that **Deploy to Application Server** is selected.

3. Click **Next**.

   The Deploy Configuration dialog is displayed, as shown in **Figure 5–36**.
4. Leave the default values selected and click **Next**.

The Select Server dialog is displayed, as shown in **Figure 5–37**.

**Figure 5–37  Select Server Dialog**
5. Select an available application server that was configured and click **Next**.

The SOA Servers dialog is displayed, as shown in **Figure 5–38**.

*Figure 5–38  SOA Servers Dialog*

6. Select a target SOA server and click **Next**.

The Summary dialog is displayed, as shown in **Figure 5–39**.
7. Review and verify all the available deployment information for your project and click Finish.

The process is deployed successfully, as shown in Figure 5–40.

5.2.4 Invoking the Input XML Document in the Oracle Enterprise Manager Console

Perform the following steps to invoke the input XML document in the Oracle Enterprise Manager console.

1. Log in to the Oracle Enterprise Manager console by using the following URL:
   http://localhost:7001/em

2. Expand your domain in the left pane followed by the SOA folder, as shown in Figure 5–41.
3. As shown in Figure 5–42, select an available project (for example, Siebel_Outbound_queryWithView_Mediator).

4. Click Test in the right pane.

The Test Web Service page is displayed, as shown in Figure 5–43.
5. Click the Request tab.

6. Scroll down to the Input Arguments section, as shown in Figure 5–44.

7. Select XML View from the list in the upper-left corner.


9. Click Test Web Service.

The output response is received in the Response tab of the Oracle Enterprise Manager console, as shown in Figure 5–45.
5.3 Configuring a Mediator Inbound Process

The following example describes how to configure a Mediator inbound process to your Siebel system, using a Mediator project in Oracle JDeveloper.

Prerequisites
Before you design a Mediator inbound process, you must generate the respective WSDL file using Application Explorer. For more information, see "Generating WSDL for Event Integration" on page 4-44.

5.3.1 Creating a New SOA Application for the Inbound Mediator Process

Perform the following steps to create a new SOA application for the inbound Mediator process:

1. Open Oracle JDeveloper on your system.
2. Click Application in the menu bar and select New from the menu, as shown in Figure 5–46.
Configuring a Mediator Inbound Process

**Figure 5–46** New Option

The Create SOA Application wizard is displayed, as shown in Figure 5–47.

**Figure 5–47** Create SOA Application Wizard

The Create SOA Application wizard is displayed, as shown in Figure 5–47.
3. From the Application Template list, click **SOA Application**.

4. Enter a name for the new SOA application (for example, Siebel_Inbound_Mediator) and click **Next**.

   The Name your project page is displayed, as shown in **Figure 5–48**.

**Figure 5–48 Name Your Project Page**

![Name Your Project Page](image)

5. Enter a project name (for example, SampleAccount_Receive) and click **Next**.

   The Configure SOA settings page is displayed, as shown in **Figure 5–49**.
6. From the Composite Template list, select **Empty Composite** and click **Finish** to show the Application Navigator, as shown in Figure 5–50.

![Figure 5–50 SOA Application (Siebel_Inbound_Mediator) and Associated Project (SampleAccount_Receive) Added to the Application Navigator Tab](image)

The new SOA application (Siebel_Inbound_Mediator) and associated project (SampleAccount_Receive) are added to the Application Navigator tab in the left pane.
5.3.2 Defining a Mediator Inbound Process

This section describes how to define a Mediator inbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Inbound Mediator Process Component With a File Adapter
3. Configuring the Routing Rules

Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the Third Party Adapter component from the Component Palette tab (Service Adapters section) to the Exposed Services pane, as shown in Figure 5–51.

Figure 5–51 Third Party Adapter Component Being Dragged to the Exposed Services Pane

The Create Third Party Adapter Service dialog is displayed, as shown in Figure 5–52.
2. Enter a name for the third party adapter service.
3. Ensure that Service is selected from the Type list (default).
4. Click the Find existing WSDLs icon, which is located to the right of the WSDL URL field.
   The SOA Resource Browser dialog is displayed, as shown in Figure 5–53.
5. Browse and select an inbound WSDL file from the following directory:

\<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdl

6. Click OK.

The Localize Files dialog is displayed, as shown in Figure 5–54.

**Figure 5–54 Localize Files Dialog**

```
Localize Files

file:C:\oracle\Middleware\Oracle SOA Suite\thirdparty\ApplicationAdapters\wsdl\Sample_Account_receive.wsdl

Copy Options:  \Maintain original directory structure for imported files\n
The following files will be created in directory:
C:\\Developer\\mywork\\Siebel_Inbound_SPEL\\SampleAccount_Receive :
Sample_Account_receive.wsdl
Sample_Account_receive_request.xsd
```

7. Click OK.

The inbound WSDL file and associated receive/request XML schema file (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 5–55.
8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in Figure 5–56.

![Figure 5–56 SOA Resource Browser Dialog](image)

9. Browse and select the JCA properties file from the following directory:

   `<ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/wsdl`

10. Click **OK**.
As shown in Figure 5–57, the following message is displayed.

**Figure 5–57 Copy File Confirmation Message Dialog**

![Copy File Confirmation Message Dialog](image)

The jca file Sample_Account_receive.jca is external to the current project. In order to make this file available to your project at runtime, JDeveloper can now make a local copy. Do you want to copy file to Sample_Account_receive_3P.jca?

11. Click Yes.

A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 5–58.

**Figure 5–58 Create Third Party Adapter Service Dialog**

![Create Third Party Adapter Service Dialog](image)

12. Click OK.

The third party adapter service component (SampleAccount) is created and displayed in the Exposed Services pane, as shown in Figure 5–59.
Configuring an Inbound Mediator Process Component With a File Adapter

Perform the following steps to configure an inbound Mediator process component with a File adapter.

1. Drag and drop the Mediator component from the Component Palette tab (Service Components section) to the Components pane, as shown in Figure 5–60.

The Create Mediator dialog is displayed, as shown in Figure 5–61.
Figure 5–61  Create Mediator Dialog

2. In the Name field, enter a name to identify the new inbound Mediator process component (for example, SampleAccount_Mediator).

3. From the Template list, select Define Interface Later.

4. Click the OK.

   The new Mediator process component is added to the Components pane, as shown in Figure 5–62.

Figure 5–62  Mediator Process Component Being Added to the Components Pane

5. Drag and drop the File Adapter component from the Service Adapters pane to the External References pane.

   The Adapter Configuration Wizard - Welcome page is displayed, showing the Welcome page, as shown in Figure 5–63.
6. Click Next.

The Service Name page is displayed, as shown in Figure 5–64.
7. Type a name for the new File adapter in the Service Name field and click **Next**. The Adapter Interface page is displayed, as shown in **Figure 5–65**.
8. Ensure that the Define from operation and schema (specified later) option is selected.

9. Click Next.

The Operation page is displayed, as shown in Figure 5–66.
10. Select **Write File** from the list of Operation Type options and specify an Operation Name (for example, Write).

11. Click **Next**.

The File Configuration page is displayed, as shown in Figure 5–67.
12. Specify a location on your file system for the output file.
13. In the File Naming Convention field, specify a name for the output file.
14. Click Next.

The Messages page is displayed, as shown in Figure 5–68.
15. Click **browse for schema file**, which is located to the right of the URL field. The Type Chooser dialog is displayed, as shown in Figure 5–69.
16. Expand Project WSDL Files, Sample_Account_receive.wsdl, Inline Schemas, and schema.

17. Select the available schema (for example, SiebelMessage)

18. Click OK.

You are returned to the Messages page, as shown in Figure 5–70.
Figure 5–70 Messages Page

19. Click Next.

The Finish page is displayed, as shown in Figure 5–71.
20. Click Finish.

21. Create a connection between the inbound Mediator process component and the third party adapter service component, as shown in Figure 5–72.

Figure 5–72  Connection Between the Inbound Mediator Process Component and the Third Party Adapter Service Component

22. Create a connection between the inbound Mediator process component and the File adapter component, as shown in Figure 5–73.
You are now ready to configure the routing rules.

**Configuring the Routing Rules**

Perform the following steps to configure routing rules for the Mediator inbound process component:

1. Double-click the inbound Mediator process component in the Components pane, as shown in Figure 5–74.

**Figure 5–74  Inbound Mediator Process Component in the Components Pane**

The Routing Rules dialog is displayed, as shown in Figure 5–75.
2. In the <<Filter Expression>> area, click the Select an existing mapper file or create a new one icon to the right of the Transform Using field.

The Request Transformation Map dialog is displayed, as shown in Figure 5–76.

3. Select the Create New Mapper File option and click OK.

The SiebelMessage_To_SiebelMessage.xsl tab is displayed, as shown in Figure 5–77.


The Auto Map Preferences dialog is displayed, as shown in Figure 5–78.
5. Retain the default values and click OK.

As shown in Figure 5–79, the mapping is now complete.

6. As shown in Figure 5–80, click the Save All icon in the menu bar to save the new inbound Mediator process component that was configured.
You are now ready to deploy the Mediator inbound process. You can follow the same procedure that is described in "Defining a BPEL Inbound Process" on page 4-54.

Once event messages are triggered through Siebel, output XML is received in the location that was specified for the File adapter component. For more information on triggering events in Siebel, see “Triggering an Event in Siebel” on page 4-67.
Integration With BPM Service Components in the Oracle SOA Suite

Oracle Application Adapter for Siebel integrates seamlessly with Oracle Business Process Management (BPM) to facilitate Web service integration. Oracle BPM is based on the Service-Oriented Architecture (SOA). It consumes adapter services exposed as Web Service Definition Language (WSDL) documents.

This chapter contains the following topics:

- Section 6.1, "Overview"
- Section 6.2, "Deployment of Adapter"
- Section 6.3, "Configuring a New Application Server Connection"
- Section 6.4, "Designing an Outbound BPM Process Using Transformations for Service Integration"
- Section 6.5, "Designing an Inbound BPM Process Using Transformations for Event Integration"

6.1 Overview

To integrate with Oracle BPM, Oracle Application Adapter for Siebel must be deployed in the same WLS container as Oracle BPM. The underlying adapter services must be exposed as WSDL files, which are generated during design time in Oracle Adapter Application Explorer (Application Explorer) for both request-response (outbound) and event notification (inbound) services of the adapter. For more information, see "Generating WSDL (J2CA Configurations Only)" on page 2-28.

The generated WSDL files are used to design the appropriate BPM processes for inbound or outbound adapter services. A completed BPM process must be successfully compiled in JDeveloper and deployed to a BPM server. Upon deployment to the BPM server, every newly built process is automatically deployed to the Oracle Enterprise Manager console, where you run, monitor, and administer BPM processes, and listen to adapter events.

6.2 Deployment of Adapter

During installation, Oracle Application Adapter for Siebel is deployed as a J2CA 1.0 resource adapter within the WLS container. The adapter must be deployed in the same WLS container as Oracle BPM.
6.3 Configuring a New Application Server Connection

For more information on how to configure a new Application Server connection in Oracle JDeveloper, see Section 4.3, "Configuring a New Application Server Connection".

6.4 Designing an Outbound BPM Process Using Transformations for Service Integration

The following tools are required to complete your outbound design-time configuration:

- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPM Designer (JDeveloper) or Eclipse

**Note:** The examples in this chapter demonstrate the use of JDeveloper.

Before you design a BPM process, you must generate the respective WSDL file using Application Explorer. For more information, see "Siebel Service Integration" on page 4-8.

6.4.1 Creating an Empty Composite for BPM

Perform the following steps to create an empty composite for BPM:

1. Create a new BPM application.
2. Enter a name for the new BPM application and click **Next**, as shown in Figure 6–1.
3. Enter a project name (for example, Siebel_JCA_Outbound) and click Next.
The Configure SOA settings page is displayed, as shown in Figure 6–3.

**Figure 6–3 Configure SOA Settings Page**

4. From the Composite Template list, select **Empty Composite** and click **Finish**.

### 6.4.2 Defining a BPM Outbound Process

This section describes how to define a BPM outbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Outbound BPM Process Component
3. Creating a File Adapter for the Write Operation

**Configuring a Third Party Adapter Service Component**

Perform the following steps to create a third party adapter service component:

1. As shown in Figure 6–4, drag and drop the **Third Party Adapter** component from the Service Adapters pane to the External References pane.
Figure 6–4 Third Party Adapter Component Being Dragged from the Service Adapters Pane to the External References Pane

![Diagram of Third Party Adapter Component](image)

The Create Third Party Adapter Service dialog is displayed, as shown in Figure 6–5.

Figure 6–5 Create Third Party Adapter Service Dialog

![Create Third Party Adapter Service Dialog](image)

2. Enter a name for the third party adapter service.
3. Ensure that Reference is selected from the Type list (default).
4. Click the Find existing WSDLs icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed, as shown in Figure 6–6.
5. Browse and select an outbound Siebel WSDL file from the following directory:
   <ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls

6. Click OK.
   The Localize Files dialog is displayed, as shown in Figure 6–7.
7. Click OK.

The outbound WSDL file and associated request and response XML schema files (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 6–8.
8. Click the **Find JCA file** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in **Figure 6–9**.

**Figure 6–9  SOA Resource Browser Dialog**

9. Browse and select the JCA properties file from the following directory:

   `<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls`

10. Click **OK**.

    As shown in **Figure 6–10**, the following message is displayed.

**Figure 6–10  Copy File Confirmation Message**

11. Click **Yes**.

    A copy of the JCA properties file is made in the project folder.
You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 6–11.

**Figure 6–11  Create Third Party Adapter Service Dialog**

![Create Third Party Adapter Service Dialog](image)

12. Click OK.

   The third party adapter service component (QueryWithView) is created and displayed in the External References pane, as shown in Figure 6–12.

**Figure 6–12  Third Party Adapter Service Component**

![Third Party Adapter Service Component](image)
You are now ready to configure an outbound BPM process component.

### Configuring an Outbound BPM Process Component

Perform the following steps to configure an outbound BPM process component:

1. Drag and drop the **BPMN Process** component from the Service Components pane to the Components pane, as shown in **Figure 6–13**.

![Figure 6–13 BPMN Process Component Being Dragged to the Components Pane](image)

2. Accept the default option that is selected in the left pane (Default Process) and click **Finish**.

   The BPMN Process component is created, as shown in **Figure 6–15**.

![Figure 6–14 Create BPMN Process Dialog](image)
3. Double-click the BPMN Process component in the Components pane. The BPMN process is displayed, as shown in Figure 6–16.

4. Click the Activity icon, as shown in Figure 6–17.
5. As shown in Figure 6–18, drop the Activity icon on the wire between the Start and End event components.

The Properties - Activity dialog is displayed, as shown in Figure 6–19.
6. Click the **Implementation** tab, as shown in Figure 6–20.

7. Select **Service Task** from the Implementation Type list, as shown in Figure 6–21.
8. Select **Service Call** from the Implementation list.

9. Click the Browse icon to the right of the Name field, as shown in Figure 6–22.

The Type dialog is displayed, as shown in Figure 6–23.
10. Select the Third Party Service that has been created (for example, QueryWithView) and click OK.

You are returned to the Properties - Activity dialog, as shown in Figure 6–24.

11. Select the Use Associations option that is located under Data Associations.

12. Click the Edit Data Associations icon.

The Data Associations dialog is displayed, as shown in Figure 6–25.
13. Right-click the Process node in the right pane and select Add.
   The Create Data Object dialog is displayed, as shown in Figure 6–26.

14. Enter a name in the Name field (for example, Request) and click Browse more types to the right of the Type field.
   The Browse Types dialog is displayed, as shown in Figure 6–27.
15. As shown in Figure 6–28, select `<Component>` from the Type list.

16. Select the first component (for example, Siebel) and click OK.

You are returned to the Create Data Object dialog, as shown in Figure 6–29.
17. Click **OK**.

The Data Object (for example, Request) that has been created is displayed under the Process node in the Data Associations dialog, as shown in Figure 6–30.

18. Create another Data Object by right-clicking the **Process** node in the right pane and selecting **Add**, as shown in Figure 6–31.

The Create Data Object dialog is displayed, as shown in Figure 6–32.
19. Enter a name in the Name field (for example, Response) and click **Browse more types** to the right of the Type field.

   The Browse Types dialog is displayed, as shown in Figure 6–33.

20. As shown in Figure 6–34, select `<Component>` from the Type list.
21. Select the second component (for example, SiebelResponse) and click **OK**.

   You are returned to the Create Data Object dialog, as shown in **Figure 6–35**.

22. Click **OK**.

   The Data Object (for example, Response) that has been created is displayed under the Process node in the Data Associations dialog, as shown in **Figure 6–36**.
23. Select the Request Data Object under the Process node in the right pane and drag it to the field below the Inputs area, as shown in Figure 6–37.

Figure 6–37  Data Associations Dialog

24. Select the Response Data Object under the Process node in the right pane and drag it to the field below the Outputs area, as shown in Figure 6–38.

Figure 6–38  Data Associations Dialog

25. Click OK.

You are returned to the Properties - Activity dialog, as shown in Figure 6–39.
26. Click **OK**.

As shown in **Figure 6–40**, the Service Task is created between the Start and End Event components.

27. Save the process and double-click the Start event component.

The Properties - Start dialog is displayed, as shown in **Figure 6–41**.
28. Click the **Implementation** tab, as shown in **Figure 6–42**.

**Figure 6–42  Implementation Tab**

29. Click the **Plus** icon to the right of the Arguments Definition field.
   
   The Create Argument dialog is displayed, as shown in **Figure 6–43**.
30. Enter a name in the Name field (for example, argument1) and click **Browse more types** to the right of the Type field.

   The Browse Types dialog is displayed, as shown in **Figure 6–44**.

   **Figure 6–44   Browse Types Dialog**

31. Select `<Component>` from the Type list, as shown in **Figure 6–45**.

   **Figure 6–45   Browse Types Dialog**
32. Select the first component (for example, Siebel) and click OK.
You are returned to the Create Argument dialog, as shown in Figure 6–46.

![Create Argument Dialog](image)

**Figure 6–46  Create Argument Dialog**

33. Click OK.
You are returned to the Properties - Start dialog, as shown in Figure 6–47.

![Properties - Start Dialog](image)

**Figure 6–47  Properties - Start Dialog**

34. Select the **Use Associations** option that is located under Data Associations.

35. Click the **Edit Data Associations** icon.
The Data Associations dialog is displayed.

36. As shown in Figure 6–48, select the Request Data Object under the Process node in the right pane and drag it to the field below the Outputs area.
37. Click OK.

You are returned to the Properties - Start dialog, as shown in Figure 6–49.

38. Click OK.

You are returned to the Process workspace area.

39. As shown in Figure 6–50, double-click composite.xml in the left pane.
40. Click the Save All icon in the menu bar to save the new outbound BPM process component that was configured.

You are now ready to create a File adapter for the write operation.

Creating a File Adapter for the Write Operation
Perform the following steps to create a File adapter for the write operation:

1. Drag and drop the File Adapter component from the Service Adapters pane to the External References pane, as shown in Figure 6–51.

Figure 6–51 File Adapter Component Selected in the Service Adapters Pane

The Adapter Configuration Wizard is displayed, as shown in Figure 6–52.
2. Click Next.

The Service Name page is displayed, as shown in Figure 6–53.
3. Type a name for the new File adapter in the Service Name field and click **Next**. The Adapter Interface page is displayed, as shown in Figure 6–54.
4. Ensure that the Define from operation and schema (specified later) option is selected.

5. Click Next.

The Operation page is displayed, as shown in Figure 6–55.
6. Select **Write File** from the list of Operation Type options and specify an Operation Name (for example, Write).

7. Click **Next**.

The File Configuration page is displayed, as shown in **Figure 6–56**.
8. Specify a location on your file system for the output file.

9. In the File Naming Convention field, specify a name for the output file.

10. Click Next.

   The Messages page is displayed, as shown in Figure 6–57.
11. Click browse for schema file, which is located to the right of the URL field. The Type Chooser dialog is displayed, as shown in Figure 6–58.

13. Select the available schema (for example, SiebelResponse)

14. Click OK.

You are returned to the Messages page, as shown in Figure 6–59.

Figure 6–59  Messages Page

15. Click Next.

The Finish page is displayed, as shown in Figure 6–60.
16. Click Finish.

As shown in Figure 6–61, the File Adapter service is created in the External References pane.

17. Double-click the BPMN Process component.
The BPMN process is displayed, as shown in Figure 6–62.

**Figure 6–62 Activity Icon**

18. Click the Activity icon.

19. Drop the Activity icon on the wire between the Service Task and End event components, as shown in Figure 6–63.

**Figure 6–63 Activity Icon Placed on the Wire Between the Service Task and End Event Components**

The Properties - Activity dialog is displayed, as shown in Figure 6–64.
20. As shown in Figure 6–65, click the **Implementation** tab.

21. As shown in Figure 6–66, select **Service Task** from the Implementation Type list.
22. Select **Service Call** from the Implementation list.

23. Click the **Browse** icon to the right of the Name field, as shown in Figure 6–67.

The Type dialog is displayed, as shown in Figure 6–68.
24. Select the service for write operation that has been created (for example, Siebel_File_Adapter) and click OK.

You are returned to the Properties - Activity dialog, as shown in Figure 6–69.

25. Select the **Use Transformations** option that is located under Data Associations.

26. Click the **Edit XSL Transformations** icon.

The Transformations dialog is displayed, as shown in Figure 6–70.
27. Click the **Plus** icon.

The Transformation dialog is displayed, as shown in **Figure 6–71**.

**Figure 6–71  Transformation Dialog**

28. As shown in **Figure 6–72**, select the **Response** Data Object that was created from the Sources list.
29. Click the Plus icon so that the Data Object 'Response' is added to the Selected Elements area.

30. Accept the default value that is selected in the Target list, as shown in Figure 6–73.
31. Enter a name (for example, transformation) in the Create field and click OK. As shown in Figure 6–74, you are returned to the Transformations dialog.

32. Click OK.

You are returned to the Properties - Activity dialog, as shown in Figure 6–75.
33. Click OK.

The transformation.xsl tab is displayed, as shown in Figure 6–76.

Figure 6–76 The transformation.xsl Tab

34. Automap the Source and Target elements.

The Auto Map Preferences dialog is displayed, as shown in Figure 6–77.
35. Accept the default values and click **OK**.

The transformation is completed, as shown in Figure 6–78.

36. Save the transformation.

37. Return to the Process workspace area, as shown in Figure 6–79.
The ServiceTask1 component is created between the ServiceTask component and the End event component.

38. Save the process.

39. Double-click the End event component.

   The Properties - End dialog is displayed.

40. As shown in Figure 6–80, click the Implementation tab.

**Figure 6–80  Properties - End Dialog**

41. Select None from the Implementation Type list, as shown in Figure 6–81.
42. Click OK.

You are returned to the Process workspace area, as shown in Figure 6–82.

![Figure 6–81 None Option in Implementation Type List](image)

43. Save the Process and double-click the composite.xml file.

44. Click the Save All icon in the menu bar to save the new outbound BPM process component that was configured.

You are now ready to deploy the outbound BPM process.
6.4.3 Deploying the BPM Outbound Process

Perform the following steps to deploy the BPM outbound process.

1. Right-click the project name in the left pane (for example, Siebel_JCA_Outbound), select Deploy, and then click Siebel_JCA_Outbound, as shown in Figure 6–83.

Figure 6–83 Deploy Option

The Deployment Action page is displayed.

2. Ensure that Deploy to Application Server is selected.

3. Click Next.

The Deploy Configuration page is displayed, as shown in Figure 6–84.
4. Leave the default values selected and click Next.

The Select Server page is displayed, as shown in Figure 6–85.

**Figure 6–85 Select Server Page**
5. Select an available application server that was configured and click **Next**.
   
The SOA Servers page is displayed, as shown in **Figure 6–86**.

**Figure 6–86  SOA Servers Page**

6. Select a target SOA server and click **Next**.
   
The Summary page is displayed, as shown in **Figure 6–87**.

**Figure 6–87  Summary Page**
7. Review and verify all the available deployment information for your project and click Finish.

   The process is deployed successfully, as shown in Figure 6–88.

   **Figure 6–88 Deployment Successful Message**

   ![Deployment Successful Message](image)

6.4.4 Invoking the Input XML Document in the Oracle Enterprise Manager Console

   Perform the following steps to invoke the input XML document in the Oracle Enterprise Manager console.

1. Logon to the Oracle Enterprise Manager console, as shown in Figure 6–89.

   **Figure 6–89 Oracle Enterprise Manager Console**

   ![Oracle Enterprise Manager Console](image)

2. Expand your domain in the left pane followed by the SOA folder.

3. Select an available project (for example, Siebel_JCA_Outbound).

4. Click Test.

5. Click the Request tab.

6. Provide an appropriate input value in the Value field and click Test Web Service.

   A response is received in the Response tab to indicate that invocation was successful in the Oracle Enterprise Manager console.

7. Navigate to the defined output directory on your file system and open the XML response document that was received.

   The XML response document contains the generated output with values.
6.5 Designing an Inbound BPM Process Using Transformations for Event Integration

This section demonstrates how Oracle Application Adapter for Siebel integrates with Siebel to receive event data.

The following tools are required to complete your adapter design-time configuration:
- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPM Designer (JDeveloper) or Eclipse

**Note:** The examples in this chapter demonstrate the use of JDeveloper.

Before you design a BPM process, you must generate the respective WSDL file using Application Explorer. For more information, see Generating WSDL for Event Integration on page 4-44.

6.5.1 Creating an Empty Composite for BPM

Perform the following steps to create an empty composite for BPM:

1. Create a new BPM application.
2. Enter a name for the new BPM application and click Next, as shown in Figure 6–90.

![Figure 6–90 Name Your Application Page](image)

The Name your project page is displayed, as shown in Figure 6–91.
3. Enter a project name (for example, Siebel_JCA_Inbound) and click Next.

The Configure SOA settings page is displayed, as shown in Figure 6–92.
4. From the Composite Template list, select Empty Composite and click Finish.

6.5.2 Defining a BPM Inbound Process

This section describes how to define a BPM inbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Inbound BPM Process Component
3. Creating a File Adapter for the Write Operation

Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the Third Party Adapter component from the Service Adapters pane to the Exposed References pane, as shown in Figure 6–93.

![Figure 6–93 Third Party Adapter Component Being Dragged to the Exposed References Pane](image)

The Create Third Party Adapter Service dialog is displayed, as shown in Figure 6–94.
2. Enter a name for the third party adapter service.
3. Ensure that Service is selected from the Type list (default).
4. Click the Find existing WSDLs icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed, as shown in Figure 6–95.
5. Browse and select an inbound WSDL file from the following directory:
   `<ADAPTER_HOME>/soa/thirdparty/ApplicationAdapters/wsdl`

6. Click **OK**.
   The Localize Files dialog is displayed, as shown in **Figure 6–96**.
7. Click OK.

The inbound WSDL file and associated receive_request XML schema file (.xsd) are imported to the project folder that has been created.

As shown in Figure 6–97, you are returned to the Create Third Party Adapter Service dialog.

Figure 6–97 Create Third Party Adapter Service Dialog

8. Click the Find JCA file icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed, as shown in Figure 6–98.
9. Browse and select the JCA properties file from the following directory:

<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\wsdls

10. Click OK.

The following message is displayed, as shown in Figure 6–99.

![Copy File Confirmation Message](image)

11. Click Yes.

A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog, as shown in Figure 6–100.
Figure 6–100  Create Third Party Adapter Service Dialog

12. Click OK.

The third party adapter service component (SampleAccount) is created in the Exposed References pane, as shown in Figure 6–101.

Figure 6–101  Third Party Adapter Service Component Created in the Exposed References Pane
You are now ready to configure an inbound BPM process component.

**Configuring an Inbound BPM Process Component**

Perform the following steps to configure an inbound BPM process component:

1. As shown in Figure 6–102, drag and drop the BPMN Process component from the Service Components pane to the Components pane.

![Figure 6–102 BPMN Process Component Being Dragged to the Components Pane](image1)

The Create BPMN Process dialog is displayed, as shown in Figure 6–103.

![Figure 6–103 Create BPMN Process Dialog](image2)

2. Select the From Pattern option in the left pane and ensure that Manual Process is selected in the Patterns area.

3. Click Finish.

The BPMN Process component is created, as shown in Figure 6–104.

![Figure 6–104 BPMN Process Component Created](image3)
4. Double-click the BPMN Process component in the Components pane. The BPMN process is displayed, as shown in Figure 6–105.

5. Right-click **UserTask** and select **Delete** from the menu.
6. As shown in Figure 6–106, double-click the **Start** event component.
The Properties - Start dialog is displayed, as shown in Figure 6–107.

7. Click the Implementation tab, as shown in Figure 6–108.
8. Select **Message** from the Implementation Type list, as shown in Figure 6–109.

9. Select **Interface from Catalog** from the Implementation list.

10. Click the **Browse** icon to the right of the Name field, as shown in Figure 6–110.
11. Select the Third Party Service that has been created and click **OK**.

You are returned to the Properties - Start dialog, as shown in **Figure 6–112**.
12. Select the **Use Associations** option that is located under Data Associations.

13. Click the **Edit Data Associations** icon.

   The Data Associations dialog is displayed, as shown in **Figure 6–113**.

**Figure 6–113  Data Associations Dialog**

14. Right-click the **Process** node in the right pane and select **Add**.

   The Create Data Object dialog is displayed, as shown in **Figure 6–114**.
15. Enter a name in the Name field (for example, sample_account_in) and click Browse more types to the right of the Type field.

The Browse Types dialog is displayed, as shown in Figure 6–115.

16. As shown in Figure 6–116, select <Component> from the Type list.
17. Select the first component (for example, SiebelMessage) and click **OK**.

You are returned to the Create Data Object dialog, as shown in **Figure 6–117**.

18. Click **OK**.

The Data Object (for example, sample_account_in) that has been created is displayed under the Process node in the Data Associations dialog, as shown in **Figure 6–118**.
19. As shown in Figure 6–119, select the \texttt{sample\_account\_in} Data Object under the Process node in the right pane and drag it to the field below the Outputs area.

20. Click OK.

You are returned to the Properties - Start dialog, as shown in Figure 6–120.
21. Click OK.
You are returned to the Process workspace area.

22. Double-click composite.xml in the left pane, as shown in Figure 6–121.

23. Click the Save All icon in the menu bar to save the new inbound BPM process component that was configured.
You are now ready to create a File adapter for the write operation.

Creating a File Adapter for the Write Operation
Perform the following steps to create a File adapter for the write operation:
1. Drag and drop the **File Adapter** component from the Service Adapters pane to the External References pane, as shown in Figure 6–122.

![Figure 6–122 File Adapter Component Being Dragged to External References Pane](image)

The Adapter Configuration Wizard is displayed, as shown in Figure 6–123.

![Figure 6–123 Adapter Configuration Wizard](image)

This wizard helps you create a service using the File Adapter. You will be asked to specify configuration parameters and define an operation for the service.

2. Click **Next**.

The Service Name page is displayed, as shown in Figure 6–124.
3. Type a name for the new File adapter in the Service Name field and click **Next**. The Adapter Interface page is displayed, as shown in Figure 6–125.
4. Ensure that the **Define from operation and schema (specified later)** option is selected.

5. Click **Next**.

   The Operation page is displayed, as shown in Figure 6–126.
6. Select **Write File** from the list of Operation Type options and specify an Operation Name (for example, Write).

7. Click **Next**.

The File Configuration page is displayed, as shown in Figure 6–127.
8. Specify a location on your file system for the output file.
9. In the File Naming Convention field, specify a name for the output file.
10. Click Next.

The Messages page is displayed, as shown in Figure 6–128.
11. Click **browse for schema file**, which is located to the right of the URL field. The Type Chooser dialog is displayed, as shown in **Figure 6–129**.

**Figure 6–129  Type Chooser Dialog**
12. Expand **Project Schema Files** and then **Sample_Account_receive_request.xsd**.

13. Select the available schema (for example, SiebelMessage)

14. Click **OK**.
   
   You are returned to the Messages page, as shown in **Figure 6–130**.

---

**Figure 6–130 Messages Page**

15. Click **Next**.
   
   The Finish page is displayed, as shown in **Figure 6–131**.
16. Click Finish.

As shown in Figure 6–132, the File Adapter service is created in the External References pane.

17. Double-click the BPMN Process component.

The BPMN process is displayed, as shown in Figure 6–133.
18. Click the Activity icon.

19. As shown in Figure 6–134, drop the Activity icon on the wire between the Start and End event components.

The Properties - Activity dialog is displayed, as shown in Figure 6–135.
20. Click the Implementation tab, as shown in Figure 6–136.

21. Select Service Task from the Implementation Type list, as shown in Figure 6–137.
22. Select **Service Call** from the Implementation list.

23. Click the **Browse** icon to the right of the Name field, as shown in Figure 6–138.

The Type dialog is displayed, as shown in Figure 6–139.
24. Select the service for write operation that has been created and click OK.
You are returned to the Properties - Activity dialog, as shown in Figure 6–140.

25. Select the Use Transformations option that is located under Data Associations.
26. Click the Edit XSL Transformations icon.
The Transformations dialog is displayed, as shown in Figure 6–141.
27. Click the **Plus** icon.

   The Transformation dialog is displayed, as shown in Figure 6–142.

**Figure 6–142 Transformation Dialog**

28. Select the **sample_account_in** Data Object that was created from the Sources list.

29. Click the **Plus** icon so that the Data Object ‘sample_account_in’ is added to the Selected Elements area, as shown in Figure 6–143.
30. As shown in Figure 6–144, accept the default value that is selected in the Target list.

**Figure 6–143 Transformation Dialog**

**Figure 6–144 Transformation Dialog**
31. Enter a name (for example, sample) in the Create field and click **OK**.

You are returned to the Transformations dialog, as shown in **Figure 6–145**.

**Figure 6–145  Transformations Dialog**

![Transformations Dialog](image)

32. Click **OK**.

You are returned to the Properties - Activity dialog, as shown in **Figure 6–146**.
33. Click **OK**.

The sample.xsl tab is displayed, as shown in Figure 6–147.

**Figure 6–147  The sample.xsl Tab**

34. Automap the Source and Target elements.

The Auto Map Preferences dialog is displayed, as shown in Figure 6–148.
35. Accept the default values and click OK.

The transformation is completed, as shown in Figure 6–149.

36. Save the transformation.

37. As shown in Figure 6–150, return to the Process workspace area.
The ServiceTask component is created between the Start event component and the End event component.

38. Save the Process and double-click the composite.xml file.

39. Click the Save All icon in the menu bar to save the new inbound BPM process component that was configured.

You are now ready to deploy the inbound BPM process.

6.5.3 Deploying the BPM Inbound Process

Perform the following steps to deploy the BPM inbound process.

1. Right-click the project name in the left pane (for example, Siebel_JCA_Inbound), select Deploy, and then click Siebel_JCA_Inbound, as shown in Figure 6–151.

The Deployment Action page is displayed, as shown in Figure 6–152.
2. Ensure that **Deploy to Application Server** is selected.

3. Click **Next**.

   The Deploy Configuration page is displayed, as shown in Figure 6–153.
4. Leave the default values selected and click Next. The Select Server page is displayed, as shown in Figure 6–154.

**Figure 6–154 Select Server Page**
5. Select an available application server that was configured and click **Next**.

   The SOA Servers page is displayed, as shown in **Figure 6–155**.

**Figure 6–155   SOA Servers Page**

6. Select a target SOA server and click **Next**.

   The Summary page is displayed, as shown in **Figure 6–156**.

**Figure 6–156   Summary Page**
7. Review and verify all the available deployment information for your project and click Finish.

The process is deployed successfully, as shown in Figure 6–157.

**Figure 6–157 Deployment Successful Message**

---

### 6.5.4 Triggering an Event in Siebel

The following topic describes how to trigger an event in Siebel and verify event integration using Oracle Application Adapter for Siebel.

**Triggering a Siebel Event to Test Event Runtime Integration**

To trigger an event in Siebel:

1. As shown in Figure 6–158, start the Siebel Call Center by entering the following URL in a browser:

   ```
   http://host name/callcenter/start.swe
   ```

   **Figure 6–158 Site Map Option**

2. Click View and select Site Map from the list.

   The Site Map view is displayed, as shown in Figure 6–159.
3. Click **Siebel Workflow Administration**.

   The Siebel Workflow Administration page is displayed, as shown in Figure 6–160.

4. Click **Workflow Processes**.

   The Workflow Processes page is displayed, as shown in Figure 6–161.
5. Click **Query** to search for the Workflow needed to trigger a Siebel event, as shown in Figure 6–162.
6. As shown in Figure 6–163, enter a Siebel workflow name and click Search.

![Figure 6–163  Workflow Processes Page](image)

7. Select the workflow, as shown in Figure 6–164.

![Figure 6–164  Process Designer Tab](image)

8. Click the Process Designer tab and double-click the Send Siebel Quote Data HTTP workflow element.

   As shown in Figure 6–165, the Input Arguments tab is displayed.

![Figure 6–165  Input Arguments Tab](image)

9. Enter the IP address and port for the HTTPRequestURLTemplate input argument.

10. Click Return To Designer, as shown in Figure 6–166.
11. Click the Process Simulator tab, as shown in Figure 6–167.

Figure 6–167  Process Simulator tab

The Simulator tab is displayed, as shown in Figure 6–168.

Figure 6–168  Simulator Tab

12. Click Start then Continue to complete the Siebel event triggering process.

Verifying the Results
To verify your results:

1. Log in to the Oracle Enterprise Manager console by using the following URL:
2. Expand your domain in the left pane followed by the SOA folder.

3. As shown in Figure 6–169, select an available project (for example, Siebel_JCA_Inbound).

**Figure 6–169  Instances Tab**

4. Click the Instances tab in the right pane.

Recently received run-time events are displayed in the Instances tab, as shown in Figure 6–172.

**Figure 6–170  Recently Received Run-time Events in Instances Tab**

5. Select a Siebel instance ID.

The Flow Trace page is displayed, as shown in Figure 6–171.

**Figure 6–171  Flow Trace Page**

6. Select a component instance to view its detailed audit trail.
The Instance page for the selected component is displayed, as shown in Figure 6–172.

**Figure 6–172 Instance Page**

The message received from the Siebel system is displayed in the Audit Trail tab. The document tree is shown below.

```xml
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:Siebel="http://www.oracle.com/Siebel">
  <soap:Body>
    <Siebel:RequestMessage>
      <Siebel:InputVariable>
        <part name="event_sampacc_ionode">
          <Siebel:Message IntObjectFormat="Siebel Hierarchical">
            <Siebel:MessageName>Simple Account</Siebel:MessageName>
            <Siebel:MessageId>1-1C6VM</Siebel:MessageId>
            <Siebel:MessageTypes>IntegrationObject</Siebel:MessageTypes>
          </Siebel:Message>
          <Siebel:ListOfSampleAccount>
            <Siebel:Account>
              <Siebel:Culture>
              </Siebel:Culture>
              <Siebel:Description/>
              <Siebel:Division/>
              <Siebel:HomePage/>
              <Siebel:IntegrationId/>
              <Siebel:IsEmptyDatabase/>
            </Siebel:Account>
          </Siebel:ListOfSampleAccount>
        </part>
      </Siebel:InputVariable>
    </Siebel:RequestMessage>
  </soap:Body>
</soap:Envelope>
```

7. Click the **Audit Trail** tab to view the event message.

The message received from the Siebel system is displayed in the Audit Trail tab.

8. Navigate to the defined output directory on your file system and open the event messages that were created.

The event messages contain the generated output with values.
Oracle Application Adapter for Siebel integrates seamlessly with Oracle Service Bus (OSB) to facilitate Web service integration. OSB is based on the Service-Oriented Architecture (SOA). It consumes adapter services exposed as Web Service Definition Language (WSDL) documents.

This chapter contains the following topics:

- Section 7.1, "Overview of Application Adapter Integration with Oracle Service Bus"
- Section 7.2, "Configuring Outbound Processing Using Oracle Service Bus (J2CA Configuration)"
- Section 7.3, "Configuring Inbound Processing Using Oracle Service Bus (J2CA Configuration)"
- Section 7.4, "Configuring Outbound Processing Using Oracle Service Bus (BSE Configuration)"

7.1 Overview of Application Adapter Integration with Oracle Service Bus

To integrate with Oracle Service Bus (OSB), Oracle Application Adapter for Siebel must be deployed in the same Oracle WebLogic Server as OSB. The underlying adapter services must be exposed as WSDL files, which are generated during design time in Oracle Adapter Application Explorer (Application Explorer) for both request-response (outbound) and event notification (inbound) services of the adapter.

7.2 Configuring Outbound Processing Using Oracle Service Bus (J2CA Configuration)

This section describes how to configure outbound processing using Oracle Service Bus for J2CA configurations.

This section includes the following topics:

- Section 7.2.1, "Starting Oracle Service Bus and Creating Project Folders"
- Section 7.2.2, "Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus"
- Section 7.2.3, "Publishing a WSDL From Application Explorer to Oracle Service Bus"
- Section 7.2.4, "Configuring a File Type Business Service"
7.2.1 Starting Oracle Service Bus and Creating Project Folders

Perform the following steps to start Oracle Service Bus and create project folders:

1. Start the Oracle WebLogic Server for the Oracle WebLogic Server domain that you have configured.

2. Open the Oracle Service Bus Console in a Web browser by entering the following URL:

   http://hostname:port/sbconsole

   Where hostname is the name of the machine where Oracle WebLogic Server is running and port is the port for the domain you are using. The port for the default domain is 7001.

   The Oracle Service Bus Console logon page is displayed.

3. Log on to the Oracle Service Bus Console using a valid user name and password.

   The Oracle Service Bus Console home page is displayed, as shown in Figure 7–1.

   ![Figure 7–1 Oracle Service Bus Console Home Page](image)

4. Click Create in the Change Center area to start a new Oracle Service Bus session, as shown in Figure 7–2.
5. Click **Project Explorer** in the left pane, as shown in Figure 7–3.

**Figure 7–3 Project Explorer Option**

The Project Explorer page is displayed, as shown in Figure 7–4.

**Figure 7–4 Project Explorer Page**

6. Provide a valid name for the new project (for example, Siebel_Project) and click **Add Project**.

   The new project is successfully created and listed in the left pane.

7. Click the new project node in the left pane.

   The project page is displayed, as shown in Figure 7–5.
8. In the Enter New Folder Name field, type **Business Service** and click **Add Folder**.

9. In the Enter New Folder Name field, type **Proxy Service** and click **Add Folder**.

10. In the Enter New Folder Name field, type **wsdl** and click **Add Folder**.

The Business Service, Proxy Service, and wsdl folders are listed in the left pane below the new Siebel_Project node, as shown in Figure 7–6.

11. Click **Activate** in the Change Center.

   The Activate Session page is displayed, as shown in Figure 7–7.

12. Click **Submit**.

---

**Figure 7–5** The Enter New Folder Name Field

**Figure 7–6** Business Service, Proxy Service, and wsdls Folders Listed Below the Project Node

**Figure 7–7** Activate Session Page
7.2.2 Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus

Before starting and using Application Explorer to publish a WSDL directly to the Oracle Service Bus (OSB) Console (project/folder), OSB users must perform the following steps:

1. Open the command prompt window.
2. Navigate to the following directory:
   `<OSB_Home>\user_projects\domains\base_domain\bin`
3. Execute `setDomainEnv.cmd` (Windows) or `./setDomainEnv.sh` (UNIX/Linux).
   This command sets the class path for Application Explorer to access the Oracle WebLogic Server APIs to publish the WSDLs to the OSB Console.
4. Do not close the command prompt window.
5. Navigate to the following directory:
   `<OSB_Home>\3rdparty\ApplicationAdapters\tools\iwae\bin`
6. Execute `ae.bat` (Windows) or `iwae.sh` (UNIX/Linux) to start Application Explorer. You are now ready to publish WSDLs from Application Explorer to the OSB Console.

7.2.3 Publishing a WSDL From Application Explorer to Oracle Service Bus

Perform the following steps to publish a WSDL from Application Explorer to Oracle Service Bus:

1. Start Application Explorer, connect to a J2CA configuration, and connect to a Siebel target.
   For more information, see Chapter 2, "Configuring Oracle Application Server Adapter for Siebel".
2. Expand `Business Object`, `Account`, and then `Account`.
3. Right-click the `query` method and select `Create Outbound JCA Service(Request/Response)` from the menu, as shown in Figure 7–8.
The Export WSDL dialog is displayed, as shown in Figure 7–9.
In the Name field, a default file name for the WSDL file is provided. You can accept the default or provide your own.

5. Select the Export to OSB option.

6. In the Location field, enter the folder name in Oracle Service Bus where you want to publish the WSDL document.
   The location is composed of an Oracle Service Bus project name and optionally, one or more folder names. The project name and any folder names must be separated by a forward slash character “/”.

7. In the Host field, enter the name of the machine where Oracle Service Bus is installed.

8. In the Port field, enter the port that is being used by Oracle Service Bus.

9. In the User field, enter your username to access Oracle Service Bus.

10. In the Password field, enter your password to access Oracle Service Bus.

11. Click OK.
    The WSDL is published to the location specified in the Export WSDL dialog and is now available for use with a Business Service or Proxy Service in Oracle Service Bus.

12. As shown in Figure 7–10, open the Oracle Service Bus Console and select the created WSDL folder in the left pane (for example, wsdl).
13. Ensure that the exported WSDL is listed, as shown in **Figure 7–11**.

**Figure 7–11  The Exported WSDL Listed**

<table>
<thead>
<tr>
<th>Name</th>
<th>Resource Type</th>
<th>Actions</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>query_invoke</td>
<td>JCA Binding</td>
<td>aJe</td>
<td></td>
</tr>
<tr>
<td>query_invoke</td>
<td>WSDL</td>
<td>aJe</td>
<td></td>
</tr>
<tr>
<td>query_invoke_request</td>
<td>XML Schema</td>
<td>aJe</td>
<td></td>
</tr>
<tr>
<td>query_invoke_response</td>
<td>XML Schema</td>
<td>aJe</td>
<td></td>
</tr>
</tbody>
</table>

14. Click the icon that corresponds to the exported WSDL in the Actions column. The Generate WSDL and Service page is displayed, as shown in **Figure 7–12**.
Figure 7–12 Generate WSDL and Service Page

15. Provide a new WSDL name and a new service name in the corresponding fields, as shown in Figure 7–13.

Figure 7–13 Generate WSDL and Service Page

16. In the Location area, select an available project and the sub-folder that is designated for Business Services.
17. Click Generate.

If the WSDL and Service resources are successfully created, then the following message is displayed, as shown in Figure 7–14.

Figure 7–14  WSDL and Service Resources Generated Successfully Message

7.2.4 Configuring a File Type Business Service

Perform the following steps to configure a File type Business Service:

1. As shown in Figure 7–15, select the Business Service folder you created in the left pane.

Figure 7–15  Business Service Folder Selected in the Project Explorer

As shown in Figure 7–17, the General Configuration page is displayed.
3. Provide a name for the new Business Service and select **Messaging Service** from the Service Type area.

4. Click Next.

   The Message Type Configuration page is displayed, as shown in **Figure 7–18**.

---

**Figure 7–17  General Configuration Page**

<table>
<thead>
<tr>
<th>Service Name</th>
<th>Siebel_File_Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Service Type</td>
<td>Create a New Service</td>
</tr>
<tr>
<td></td>
<td>- WSDL Web Service</td>
</tr>
<tr>
<td></td>
<td>- Transport Typied Service</td>
</tr>
<tr>
<td></td>
<td>- Messaging Service</td>
</tr>
<tr>
<td></td>
<td>- Any SOAP Service</td>
</tr>
<tr>
<td></td>
<td>Create From Existing Service</td>
</tr>
<tr>
<td></td>
<td>- Business Service</td>
</tr>
<tr>
<td></td>
<td>- Proxy Service</td>
</tr>
</tbody>
</table>

---

**Figure 7–18  Message Type Configuration Page**

---

**Request Message Type**

<table>
<thead>
<tr>
<th>Option</th>
<th>Choice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Binary</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MFL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Java</td>
<td></td>
</tr>
</tbody>
</table>

**Response Message Type**

<table>
<thead>
<tr>
<th>Option</th>
<th>Choice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Binary</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Text</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MFL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XML</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Java</td>
<td></td>
</tr>
</tbody>
</table>
5. Select XML as the Request Message Type and None as the Response Message Type.

6. Click Next.

The Transport Configuration page is displayed, as shown in Figure 7–19.

Figure 7–19 Transport Configuration Page

7. Select file from the Protocol list.

8. Enter the path to a destination folder on your file system in the Endpoint URI field and click Add.

9. Click Next, as shown in Figure 7–20.

Figure 7–20 Next Button in the Transport Configuration Page

The FILE Transport Configuration page is displayed, as shown in Figure 7–21.
10. Enter the prefix and suffix for the output file to be received and click **Next**.

   The Message Handling and Advanced Settings page is displayed, as shown in Figure 7–22.

**Figure 7–22  Message Handling and Advanced Settings Page**

11. Click **Next**.

   The following Summary page is displayed, as shown in Figure 7–23.
12. Review all the information for your Business Service and click **Save**.

13. Check if the success message is displayed and click **Activate** in the Change Center, as shown in Figure 7–24.

![Figure 7–24 Active Button in the Change Center](image)

The Activate Session page is displayed, as shown in Figure 7–25.
14. Click Submit.

15. Check if the View Configuration Changes page is displayed, as shown in Figure 7–26.

### Figure 7–26  View Configuration Changes Page

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Execution Time</th>
<th>User</th>
<th>Task Status</th>
<th>Undone By</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/17/10 3:52 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/17/10 3:00 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session 1279592350918</td>
<td>weblogic</td>
<td>7/17/10 2:46 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/17/10 2:45 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/16/10 7:52 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/16/10 7:31 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session 1279592185126</td>
<td>weblogic</td>
<td>7/16/10 4:39 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/16/10 2:14 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>Activate session</td>
<td>weblogic</td>
<td>7/16/10 1:45 PM</td>
<td>weblogic</td>
<td>Activated</td>
<td>None</td>
<td></td>
</tr>
</tbody>
</table>

### 7.2.5 Configuring a Proxy Service

Perform the following steps to configure a Proxy Service:

1. As shown in Figure 7–27, select the Proxy Service folder you created in the left pane.
2. In the right pane, select **Proxy Service** from the Create Resource menu.

   The General Configuration page is displayed, as shown in **Figure 7–28**.

**Figure 7–28  General Configuration Page**

3. Provide a name for the Proxy Service and from the Service Type area select **Any XML Service**.

4. Click **Next**, as shown in **Figure 7–29**.
5. Select file from the Protocol list.
6. Enter the path to an input folder on your file system in the Endpoint URI field.
7. Click Next, as shown in Figure 7–30.

9. Click Next, as shown in Figure 7–31.

*Figure 7–31  Message Handling Page*

<table>
<thead>
<tr>
<th>Message Handling</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transaction Required</strong></td>
</tr>
<tr>
<td><strong>Same Transaction for Response</strong></td>
</tr>
<tr>
<td><strong>Content Streaming</strong></td>
</tr>
<tr>
<td><strong>Buffer Type</strong></td>
</tr>
<tr>
<td><strong>Compression</strong></td>
</tr>
</tbody>
</table>

10. Accept the default values and click Next.

The following Summary page is displayed, as shown in Figure 7–32.
11. Review all the information for your Proxy Service and click **Save**.
12. Check if the success message is displayed, as shown in Figure 7–33.

**Figure 7–33 Proxy Service Created Successfully Message**

The Service "Query_ob_Siebel" was created successfully.
7.2.6 Configuring a Pipeline

Perform the following steps to configure a Pipeline:

1. As shown in Figure 7–34, click the **Edit Message Flow** icon in the row of the Proxy Service you created.

   **Figure 7–34  Edit Message Flow Icon**

   ![Edit Message Flow Icon](image)

   The Edit Message Flow workspace area is displayed, as shown in Figure 7–35.

   **Figure 7–35  Edit Message Flow Workspace Area**

   ![Edit Message Flow Workspace Area](image)

2. Click the **Proxy Service** icon and select **Add Pipeline Pair** from the menu, as shown in Figure 7–36.

   **Figure 7–36  Add Pipeline Pair Option**

   ![Add Pipeline Pair Option](image)

3. Click the **PipelinePairNode1** icon and select **Add Route** from the menu, as shown in Figure 7–37.
4. Click the RouteNode1 icon and select Edit Route from the menu.

As shown in Figure 7–39, the Edit Stage Configuration workspace area is displayed.
5. As shown in Figure 7–40, click **Add an Action**, select **Communication** from the menu, and click **Routing**.

6. Click **<Service>**.
   The Select Service dialog is displayed, as shown in Figure 7–41.
7. Select the WSDL type Business Service configured for Siebel and click **Submit**.

As shown in **Figure 7–42**, you are returned to the Edit Stage Configuration workspace area.

**Figure 7–42  Edit Stage Configuration Workspace Area**

8. Select the name of the Siebel business object (for example, Query) as the operational attribute from the list, as shown in **Figure 7–43**.
Figure 7–43  Business Object Selected as the Operational Attribute

9. Click Save.

Figure 7–44  Add Stage Option

10. As shown in Figure 7–44, click the Response Pipeline icon and select Add Stage from the menu.

The Stage1 icon is added below the Response Pipeline icon, as shown in Figure 7–45.
11. Click the **Stage1** icon and select **Edit Stage** from the menu.

The Edit Stage Configuration workspace area is displayed, as shown in **Figure 7–46**.

**Figure 7–46  Edit Stage Configuration Workspace Area**

![Edit Stage Configuration Workspace Area](image)

12. Click **Add an Action**, select **Communication** from the menu, and click **Publish**, as shown in **Figure 7–47**.
13. Click `<Service>`.

As shown in Figure 7–48, the Select Service dialog is displayed.

14. Select a File type Business Service and click **Submit**.

You are returned to the Edit Stage Configuration workspace area, as shown in Figure 7–49.
15. Click Save.

You are returned to the Edit Message Flow workspace area, as shown in Figure 7–50.

16. Click Save.
17. As shown in Figure 7–51, check if the success message is displayed and click **Activate** in the Change Center.

The Activate Session page is displayed, as shown in Figure 7–52.

**Figure 7–52 Activate Session Page**

18. Click **Submit**.

19. As shown in Figure 7–53, check if the View Configuration Changes page is displayed.

**Figure 7–53 View Configuration Changes Page**

20. As shown in Figure 7–54, copy and paste an input XML file in the input folder you have configured.
7.3 Configuring Inbound Processing Using Oracle Service Bus (J2CA Configuration)

This section describes how to configure inbound processing using Oracle Service Bus for J2CA configurations.

This section includes the following topics:

- Section 7.3.1, "Starting Oracle Service Bus and Creating Project Folders"
- Section 7.3.2, "Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus"
- Section 7.3.3, "Publishing a WSDL From Application Explorer to Oracle Service Bus"
- Section 7.3.4, "Configuring a File Type Business Service"
- Section 7.3.5, "Configuring a Pipeline"

7.3.1 Starting Oracle Service Bus and Creating Project Folders

Perform the following steps to start Oracle Service Bus and create project folders:

1. Start the Oracle WebLogic Server for the Oracle WebLogic Server domain that you have configured.

2. Open the Oracle Service Bus Console in a Web browser by entering the following URL:

   http://hostname:port/sbconsole

   Where hostname is the name of the machine where Oracle WebLogic Server is running and port is the port for the domain you are using. The port for the default domain is 7001.
The Oracle Service Bus Console logon page is displayed.

3. Log on to the Oracle Service Bus Console using a valid user name and password.

The Oracle Service Bus Console home page is displayed, as shown in Figure 7–55.

Figure 7–55 Oracle Service Bus Console Home Page

![Figure 7–55](image)

4. As shown in Figure 7–56, click **Create** in the Change Center area to start a new Oracle Service Bus session.

Figure 7–56 Create Button in Change Center

![Figure 7–56](image)

5. Click **Project Explorer** in the left pane, as shown in Figure 7–57.

Figure 7–57 Project Explorer Option

![Figure 7–57](image)

The Project Explorer page is displayed, as shown in Figure 7–58.
6. Provide a valid name for the new project and click Add Project. The project is successfully created and listed.

7. Click on the created project.
   The project page is displayed, as shown in Figure 7–59.

8. In the Enter New Folder Name field, type Business Service and click Add Folder.

9. In the Enter New Folder Name field, type Proxy Service and click Add Folder.

10. In the Enter New Folder Name field, type wsdls and click Add Folder.

   The Business Service, Proxy Service, and wsdls folders are listed in the left pane below the project node, as shown in Figure 7–60.

7.3.2 Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus

Before starting and using Application Explorer to publish a WSDL directly to the Oracle Service Bus (OSB) Console (project/folder), OSB users must perform the following steps:

1. Open the command prompt window.
2. Navigate to the following directory:
   `<OSB_Home>`\user_projects\domains\base_domain\bin

3. Execute `setDomainEnv.cmd` (Windows) or `./setDomainEnv.sh` (UNIX/Linux).
   This command sets the class path for Application Explorer to access the Oracle
   WebLogic Server APIs to publish the WSDLs to the OSB Console.

4. Do not close the command prompt window.

5. Navigate to the following directory:
   `<OSB_Home>`\3rdparty\ApplicationAdapters\tools\iwae\bin

6. Execute `ae.bat` (Windows) or `iwae.sh` (UNIX/Linux) to start Application Explorer.
   You are now ready to publish WSDLs from Application Explorer to the OSB
   Console.

### 7.3.3 Publishing a WSDL From Application Explorer to Oracle Service Bus

This section describes how to publish a WSDL from Application Explorer to Oracle
Service Bus.

**Creating a Channel**

Before publishing a WSDL, you must first create a separate channel for every inbound
J2CA service and select that channel when you generate WSDL for inbound interaction
using Application Explorer.

To create a channel:

1. In the left pane, click **Events**.
2. Expand the **Siebel** node.
   - The ports and channels nodes appear in the left pane.

   ![Add Channel Option](image)

3. Right-click **Channels** and select **Add Channel**, as shown in **Figure 7–61**.
   - The Add Channel dialog is displayed, as shown in **Figure 7–62**.
Perform the following steps:

a. Enter a name for the channel, for example, SiebelEvent.

b. Enter a brief description.

c. From the Protocol list, select HTTP Listener, MQ Series Listener, or File Listener.

For demonstration purposes, this procedure uses the HTTP Listener as an example.

4. Click Next.

The Basic dialog is displayed, as shown in Figure 7–63.

**Figure 7–62 Add Channel Dialog**

![Add Channel Dialog](image-url)
5. Enter the system information as specified in the following table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Listener port</td>
<td>Port on which to listen for Siebel event data.</td>
</tr>
<tr>
<td>Https</td>
<td>For a secure HTTP connection, select the Https check box.</td>
</tr>
<tr>
<td>Synchronization Type</td>
<td>Select REQUEST_RESPONSE from the list, which is the recommended option.</td>
</tr>
<tr>
<td>Encoding Type</td>
<td>Choose an encoding type to be used from the list. By default, ASCII is selected.</td>
</tr>
</tbody>
</table>

6. As shown in Figure 7–64, click the PreParser tab.

7. Specify the location of the schema file that was generated for the Integration Object node using the Export Schema(s) option in Application Explorer.
8. Click OK.

The channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected, as shown in Figure 7–65.

**Figure 7–65  Disconnected Channel**

![Disconnected Channel](image)

Note: During run time, the Oracle Application Adapter for Siebel adds the namespace to the Siebel published document using the schema that is specified in the PreParser tab. If the Schema location field in the PreParser tab is left blank, then OSB processes do not work properly as the Siebel published documents do not contain any namespaces.

You must now create an Integration Object node.

**Creating an Integration Object Node**

1. Start Application Explorer.
2. Expand the Adapters node, as shown in Figure 7–66.

**Figure 7–66  Disconnected Siebel Target Node**

![Disconnected Siebel Target Node](image)

Perform the following steps:

a. Expand the Siebel node.

   The defined Siebel targets are displayed under the adapter node.

b. Click the target name, for example, siebel, under the Siebel node.

   The Connection dialog displays the values you entered.

3. Verify your connection parameters.

4. Right-click the target name and select Connect.

   The x icon disappears, indicating that the node is connected, as shown in Figure 7–67.
5. Expand the Integration Object node and select Sample Account.

6. Right-click the Sample Account node and select Add IO Node.

   The Add IO Node dialog is displayed, as shown in Figure 7–68.

7. Enter a node name (for example, Sample_Account) in the Node name field and a path to the Sample Account XSD file in the Schema location field.

   Please note:
   - **For Siebel 7.5 or later:** Generate XSD schemas directly from Siebel tools. You use the XSD schemas when you create Web services in Application Explorer. After you generate an XSD schema through Siebel tools, use it to create an IO node and a Web service.
   - **For Siebel 7.0:** You cannot generate XSD schemas directly from Siebel tools; only XDR schemas can be created. Before you create a Web service, you must first generate an XSD schema from the XDR schema using Application Explorer.

8. If the XSD schema has already been generated, then select XSD Schema. If you are using Siebel-generated XDR schemas, then do not select the XSD schema option.

9. Select a protocol (HTTP, FILE, or MQ Series) from the Protocol list.

10. Click Continue.

    The new Integration Object node is added, as shown in Figure 7–69.
Generating WSDL for Event Notification

After you create a channel and an associated Integration Object node, you must generate WSDL for the event using Application Explorer.

You must be connected to a Siebel target under the Adapters node in Application Explorer. For detailed information on how to define and connect to a target, see "Establishing a Connection (Target) for Siebel" on page 2-6.

After you connect to a Siebel target, generate WSDL for the event as follows:

1. As shown in Figure 7–70, right-click the Integration Object node (for example, Sample_Account), and then select Create Inbound JCA Service (Event).

Figure 7–70  Create Inbound JCA Service (Event) Option

The Export WSDL dialog is displayed, as shown in Figure 7–71.
2. In the Name field, a default file name for the WSDL file is provided. You can accept the default or provide your own.

3. From the Channel list, select the channel you created for this inbound service.

4. Select the **Export to OSB** option.

5. In the Location field, enter the folder name in Oracle Service Bus where you want to publish the WSDL document.

   The location is composed of an Oracle Service Bus project name and optionally, one or more folder names. The project name and any folder names must be separated by a forward slash character “/”.

6. In the Host field, enter the name of the machine where Oracle Service Bus is installed.

7. In the Port field, enter the port that is being used by Oracle Service Bus.

8. In the User field, enter your username to access Oracle Service Bus.

9. In the Password field, enter your password to access Oracle Service Bus.

10. Click **OK**.

---

**Note:** The schema validation options (Root, Namespace, Schema) are not applicable for the Oracle Application Adapter for Siebel.
The inbound WSDL is published to the location specified in the Export WSDL dialog and is now available for use with a Business Service or Proxy Service in Oracle Service Bus.

11. As shown in Figure 7–72, open the Oracle Service Bus Console and select the created WSDL folder in the left pane (for example, wsdl).

**Figure 7–72  Created WSDL Folder Selected in the Project Explorer**

![Project Explorer](image)

12. Ensure that the exported WSDL is listed, as shown in Figure 7–73.

**Figure 7–73  Exported WSDL**

![Resources](image)

13. Click the icon that corresponds to the exported WSDL in the Actions column. The Generate WSDL and Service page is displayed, as shown in Figure 7–74.
14. Provide a new WSDL name and a new service name in the corresponding fields.

15. In the Location area, select an available project and the sub-folder that is designated for Proxy Services.

16. Click Generate.

If the WSDL and Service resources are successfully created, then the following message is displayed, as shown in Figure 7–75.

Figure 7–75  WSDL and Service Resources Successfully Created Message

17. Click Proxy Service under Project Explorer and check if the generated WSDL and Proxy Service are listed, as shown in Figure 7–76.

Figure 7–76  Generated WSDL Listed
### 7.3.4 Configuring a File Type Business Service

Perform the following steps to configure a File type Business Service:

1. Select the **Business Service** folder you created in the left pane, as shown in Figure 7–77.

   ![Business Service Folder Selected in the Project Explorer](image)

2. In the right pane, select **Business Service** from the Create Resource menu, as shown in Figure 7–78.

   ![Business Service Selected from the Create Resource Menu](image)

The General Configuration page is displayed, as shown in Figure 7–79.
3. Provide a name for the Business Service and from the Service Type area select **Messaging Service**.

4. Click Next.

As shown in Figure 7–80, the Message Type Configuration page is displayed.
5. Select XML as the Request Message Type and None as the Response Message Type.

6. Click Next.

   The Transport Configuration page is displayed, as shown in Figure 7–81.

**Figure 7–81  Transport Configuration Page**

<table>
<thead>
<tr>
<th>Create a Business Service (Siebel_Project/Business Service/Siebel_Out_SampleAccount)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport Configuration</td>
</tr>
<tr>
<td>Protocol*</td>
</tr>
<tr>
<td>Load Balancing Algorithm</td>
</tr>
<tr>
<td>Endpoint URI*</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Retry Count</td>
</tr>
<tr>
<td>Retry Iteration Interval</td>
</tr>
</tbody>
</table>

7. Select file from the Protocol list.

8. Enter the path to a destination folder on your file system in the Endpoint URI field and click Add.

9. Click Next.

   The FILE Transport Configuration page is displayed, as shown in Figure 7–82.

**Figure 7–82 FILE Transport Configuration Page**

<table>
<thead>
<tr>
<th>Create a Business Service (Siebel_Project/Business Service/Siebel_Out_SampleAccount)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE Transport Configuration</td>
</tr>
<tr>
<td>Prefix</td>
</tr>
<tr>
<td>Suffix</td>
</tr>
<tr>
<td>Request encoding</td>
</tr>
</tbody>
</table>

10. Enter the prefix and suffix for the output file to be received and click Next.

   As shown in Figure 7–83, the Message Handling and Advanced Settings page is displayed.
11. Click Next.

The following Summary page is displayed, as shown in Figure 7–84.

12. Review all the information for your Business Service and click Save.

13. Check if the success message is displayed and click Activate in the Change Center, as shown in Figure 7–85.
Figure 7–85  Activate Button in the Change Center

The Activate Session page is displayed, as shown in Figure 7–86.

Figure 7–86  Activate Session Page

14. Click Submit.

15. As shown in Figure 7–87, check if the View Configuration Changes page is displayed.

Figure 7–87  View Configuration Changes Page

7.3.5 Configuring a Pipeline

Perform the following steps to configure a Pipeline:

1. As shown in Figure 7–88, click the Edit Message Flow icon in the row of the Proxy Service you created.
2. Click the displayed Proxy service icon and select **Add Route** from the menu.

   The RouteNode1 icon is added below the Request Pipeline icon.

---

**Figure 7–88   Edit Message Flow Icon**

The Edit Message Flow workspace area is displayed, as shown in **Figure 7–89**.

**Figure 7–89   Edit Message Flow Workspace Area**

2. Click the displayed Proxy service icon and select **Add Route** from the menu.

   The RouteNode1 icon is added below the Request Pipeline icon.
3. As shown in Figure 7–90, click the RouteNode1 icon and select Edit Route from the menu.

The Edit Stage Configuration workspace area is displayed, as shown in Figure 7–91.

Figure 7–91 Edit Stage Configuration Workspace Area

![Edit Stage Configuration workspace area](image)

4. Click Add an Action, select Communication from the menu, and click Routing.

Figure 7–92 Edit Stage Configuration Page

![Edit Stage Configuration page](image)

5. As shown in Figure 7–92, click <Service>.

The Select Service dialog is displayed, as shown in Figure 7–93.
6. Select a File type Business Service and click Submit.

As shown in Figure 7–94, you are returned to the Edit Stage Configuration workspace area.

7. Click Save.

As shown in Figure 7–95, you are returned to the Edit Message Flow workspace area.
8. Click Save.

9. As shown in Figure 7–96, check if the success message is displayed and click Activate in the Change Center.

The Activate Session page is displayed, as shown in Figure 7–97.

10. Click Submit.

11. As shown in Figure 7–98, check if the View Configuration Changes page is displayed.
12. Trigger an event from the Siebel system and check if the output is received in the configured output location.

7.4 Configuring Outbound Processing Using Oracle Service Bus (BSE Configuration)

This section describes how to configure outbound processing using Oracle Service Bus for BSE configurations.

This section includes the following topics:

- Section 7.4.1, "Starting Oracle Service Bus and Creating Project Folders"
- Section 7.4.2, "Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus"
- Section 7.4.3, "Publishing a WSDL From Application Explorer to Oracle Service Bus"
- Section 7.4.4, "Configuring a File Type Business Service"
- Section 7.4.5, "Configuring a WSDL Type Business Service"
- Section 7.4.6, "Configuring a Proxy Service"
- Section 7.4.7, "Configuring a Pipeline"

7.4.1 Starting Oracle Service Bus and Creating Project Folders

Perform the following steps to start Oracle Service Bus and create project folders:

1. Start the Oracle WebLogic Server for the Oracle WebLogic Server domain that you have configured.
2. Open the Oracle Service Bus Console in a Web browser by entering the following URL:

   http://hostname:port/sbconsole

   Where hostname is the name of the machine where Oracle WebLogic Server is running and port is the port for the domain you are using. The port for the default domain is 7001.

   The Oracle Service Bus Console logon page is displayed.
3. Log on to the Oracle Service Bus Console using a valid user name and password.

   The Oracle Service Bus Console home page is displayed, as shown in Figure 7–99.
4. As shown in Figure 7–100, click **Create** in the Change Center area to start a new Oracle Service Bus session.

![Create Button](image)

5. Click **Project Explorer** in the left pane, as shown in Figure 7–101.

![Project Explorer Option](image)

The Project Explorer page is displayed, as shown in Figure 7–102.
6. Provide a valid name for the new project (for example, Siebel_Project) and click **Add Project**.

   The new project is successfully created and listed in the left pane.

7. Click the new project node in the left pane.

   The project page is displayed, as shown in **Figure 7–103**.

8. In the Enter New Folder Name field, type **Business Service** and click **Add Folder**.

9. In the Enter New Folder Name field, type **Proxy Service** and click **Add Folder**.

10. In the Enter New Folder Name field, type **wsdl** and click **Add Folder**.

    The Business Service, Proxy Service, and wsdls folders are listed in the left pane below the new Siebel_Project node, as shown in **Figure 7–104**.

11. Click **Activate** in the Change Center.

    The Activate Session page is displayed, as shown in **Figure 7–105**.
7.4.2 Setting the Class Path for Application Explorer to Integrate With Oracle Service Bus

Before starting and using Application Explorer to publish a WSDL directly to the Oracle Service Bus (OSB) Console (project/folder), OSB users must perform the following steps:

1. Open the command prompt window.
2. Navigate to the following directory:
   
   `<OSB_Home>\user_projects\domains\base_domain\bin`

3. Execute `setDomainEnv.cmd` (Windows) or `. ./setDomainEnv.sh` (UNIX/Linux). This command sets the class path for Application Explorer to access the Oracle WebLogic Server APIs to publish the WSDLs to the OSB Console.
4. Do not close the command prompt window.
5. Navigate to the following directory:
   
   `<OSB_Home>\3rdparty\ApplicationAdapters\tools\iwaeb\bin`

6. Execute `ae.bat` (Windows) or `iwaeb.sh` (UNIX/Linux) to start Application Explorer. You are now ready to publish WSDLs from Application Explorer to the OSB Console.

7.4.3 Publishing a WSDL From Application Explorer to Oracle Service Bus

This section describes how to publish a WSDL from Application Explorer (BSE configuration) to Oracle Service Bus.

1. Start Application Explorer, connect to a BSE configuration, and connect to a Siebel target, as shown in Figure 7–106.
2. Expand the Business Object node and select the Account object.

3. Right-click the query method and select Create Web Service from the menu.

   The Create Web Service dialog is displayed, as shown in Figure 7–107.

   **Figure 7–107  Create Web Service Dialog**

4. Enter a service name and click Next.

5. Click OK on the next dialog that is displayed.

   Application Explorer switches the view to the Business Services node, and the new Web service appears in the left pane.
6. As shown in Figure 7–108, right-click the new Web service and select **Export WSDL** from the menu.

   The Export WSDL dialog is displayed, as shown in Figure 7–109.

7. In the **Name** field, a default file name for the WSDL file is provided. You can accept the default or provide your own.

8. In the **Location** field, enter the location where you want to publish the WSDL document.

   The location is composed of an Oracle Service Bus project name and optionally, one or more folder names. The project name and any folder names must be separated by a forward slash character “/”.

9. In the **Host** field, enter the name of the machine where Oracle WebLogic Server is running.

10. In the **Port** field, enter the port for the domain you are using.

11. In the **User** field, enter your username to access Oracle Service Bus.

12. In the **Password** field, enter your password to access Oracle Service Bus.

13. Click **OK**.

   The WSDL is published to the location specified in the Export WSDL dialog and is now available for use with a Business Service or Proxy Service in Oracle Service Bus.
7.4.4 Configuring a File Type Business Service

This section describes how to configure a File type Business Service using the Oracle Service Bus Console.

1. As shown in Figure 7–110, select the Business Service folder you created in the left pane.

![Business Service Option](image1)

2. In the right pane, select Business Service from the Create Resource menu.

The General Configuration page is displayed, as shown in Figure 7–111.

![General Configuration Page](image2)

3. Provide a name for the Business Service and from the Service Type area select Messaging Service.

4. Click Next.

The Message Type Configuration page is displayed, as shown in Figure 7–112.
5. Select XML as the Request Message Type and None as the Response Message Type.

6. Click Next.

The Transport Configuration page is displayed, as shown in Figure 7–113.

**Figure 7–113 Transport Configuration Page**

7. Select file from the Protocol list.

8. Enter the path to a destination folder on your file system in the Endpoint URI field and click Add.

9. Click Next.

The FILE Transport Configuration page is displayed, as shown in Figure 7–114.
10. Enter the prefix and suffix for the output file to be received and click **Next**.

   The Summary page is displayed, as shown in **Figure 7–115**.

**Figure 7–115  Summary Page**

11. Review all the information for your Business Service and click **Save**.

### 7.4.5 Configuring a WSDL Type Business Service

This section describes how to configure a WSDL type Business Service using the Oracle Service Bus Console.

1. As shown in **Figure 7–116**, select the **Business Service** folder you created in the left pane.
2. In the right pane, select Business Service from the Create Resource menu. The General Configuration page is displayed, as shown in Figure 7–117.

3. Provide a name for the Business Service and from the Service Type area select WSDL Web Service.

4. Click Browse.
   As shown in Figure 7–118, the Select a WSDL dialog is displayed.
5. Select a BSE WSDL that you published for Oracle Application Adapter for Siebel using Application Explorer and click **Submit**.

   The Select a WSDL Definition dialog is displayed, as shown in Figure 7–119.

   **Figure 7–119  Select a WSDL Definition Dialog**
6. Select the WSDL definition under the Ports section and click **Submit**.
   You are returned to the General Configuration page where the WSDL you selected is now available.

7. Click **Next**.
   The Transport Configuration page is displayed, as shown in **Figure 7–120**.

![Figure 7–120  Transport Configuration Page](image)

8. Leave the default values and click **Next**.
   The HTTP Transport Configuration page is displayed, as shown in **Figure 7–121**.

![Figure 7–121  HTTP Transport Configuration Page](image)

9. Leave the default values and click **Next**.
   The SOAP Binding Configuration page is displayed, as shown in **Figure 7–122**.

![Figure 7–122  SOAP Binding Configuration Page](image)
10. Click Next.

The Message Content Handling page is displayed, as shown in Figure 7–123.

**Figure 7–123 Message Content Handling Page**

11. Click Next.

The Summary page is displayed, as shown in Figure 7–124.
12. Review all the information for your Business Service and click **Save**.

### 7.4.6 Configuring a Proxy Service

This section describes how to configure a Proxy Service using the Oracle Service Bus Console.

1. As shown in **Figure 7–125**, select the **Proxy Service** folder you created in the left pane.

**Figure 7–125  Proxy Service Option**
2. In the right pane, select **Proxy Service** from the Create Resource menu. The General Configuration page is displayed, as shown in **Figure 7–126**.

**Figure 7–126  General Configuration Page**

![General Configuration Page](image)

3. Provide a name for the Proxy Service and from the Service Type area select **Any XML Service**.

4. Click **Next**. The Transport Configuration page is displayed, as shown in **Figure 7–127**.

**Figure 7–127  Transport Configuration Page**

![Transport Configuration Page](image)

5. Select **file** from the Protocol list.

6. Enter the path to an input folder on your file system in the Endpoint URI field. This is the folder where the XML input file will be copied during runtime.
7. Click Next.
   The FILE Transport Configuration page is displayed, as shown in Figure 7–128.

**Figure 7–128  FILE Transport Configuration Page**


9. Click Next.
   The Message Content Handling page is displayed, as shown in Figure 7–129.

**Figure 7–129  Message Content Handling Page**

10. Click Next.
    The Summary page is displayed, as shown in Figure 7–130.
11. Review all the information for your Proxy Service and click **Save**.

### 7.4.7 Configuring a Pipeline

This section describes how to configure a Pipeline using the Oracle Service Bus Console.

1. As shown in **Figure 7–131**, click the **Edit Message Flow** icon in the row of the Proxy Service you created.

**Figure 7–131  Edit Message Flow Icon**

The Edit Message Flow workspace area is displayed.

2. As shown in **Figure 7–132**, click the **Proxy Service** icon and select **Add Pipeline Pair** from the menu, as shown in the following image.
### Figure 7–132  Add Pipeline Pair Option

3. Click the Request Pipeline icon and select Add Stage from the menu, as shown in Figure 7–133.

### Figure 7–133  Add Stage Option

The Stage1 icon is added below the Request Pipeline icon.

### Figure 7–134  Edit Stage Option
4. As shown in Figure 7–134, click the Stage1 icon and select Edit Stage from the menu.

The Edit Stage Configuration workspace area is displayed, as shown in Figure 7–135.

Figure 7–135  Edit Stage Configuration Workspace Area

5. Click Add an Action, select Reporting from the menu, and click Log.

Figure 7–136  Edit Stage Configuration Workspace Area

6. As shown in Figure 7–136, click <Expression> to edit the expression.

The XQuery/XSLT Expression Editor is displayed, as shown in Figure 7–137.
7. In the XQuery Text area, type $inbound.

8. Click Validate and then Save.

   You are returned to the Edit Stage Configuration workspace area, as shown in Figure 7–138.

**Figure 7–138  Edit Stage Configuration Workspace Area**

9. Type any annotation/comments in the text box (for example, **********input processing***********).

10. Select Error from the severity level list.

11. Add one more Log action.

   A new Log configuration is added, as shown in the following image, as shown in Figure 7–139.
12. Click <Expression> to edit the expression.
   The XQuery/XSLT Expression Editor is displayed, as shown in Figure 7–140.

Figure 7–140  XQuery/XSLT Expression Editor

13. In the XQuery Text area, type $body.

14. Click Validate and then Save.
   You are returned to the Edit Stage Configuration workspace area, as shown Figure 7–141.
15. Type any annotation/comments in the text box (for example, **********Request Body**********).

16. Select Error from the severity level list.

17. Click Validate and then Save.

As shown in Figure 7–142, you are returned to the main Pipeline configuration area.

18. Click the PipelinePairNode1 icon and select Add Route from the menu.
19. As shown in Figure 7–143, click the RouteNode1 icon and select Edit Route from the menu.

20. As shown in Figure 7–144, click Add an Action, select Communication from the menu, and click Routing.
21. As shown in Figure 7–145, click <Service>.
   The Select Service dialog is displayed, as shown in Figure 7–146.

22. Select a WSDL type Business Service and click Submit.
   As shown in Figure 7–147, you are returned to the Edit Stage Configuration workspace area.
23. Select **query** as the operational attribute from the list.

24. Click **Validate** and then **Save**.

25. In the Response Actions area, click **Add an Action**, select **Reporting** from the menu, and click **Log**, as shown in Figure 7–148.
26. As shown in Figure 7–149, click <Expression> to edit the expression.

The XQuery/XSLT Expression Editor is displayed, as shown in Figure 7–150.

27. In the XQuery Text area, type $outbound.$

28. Click Validate and then Save.

You are returned to the Edit Stage Configuration workspace area, as shown in Figure 7–151.
Figure 7–151  Edit Stage Configuration Workspace Area

29. Type any annotation/comments in the text box (for example, **********output processing**********).

30. Select Error from the severity level list.

31. Click Validate and then Save.

    You are returned to the main Pipeline configuration area, as shown in Figure 7–152.
32. Click the **Response Pipeline** icon and select **Add Stage** from the menu. The Stage1 icon is added below the Response Pipeline icon.

33. As shown in Figure 7–153, click the **Stage1** icon and select **Edit Stage** from the menu.
The Edit Stage Configuration workspace area is displayed, as shown in Figure 7–154.

34. Click Add an Action, select Communication from the menu, and click Publish.

35. As shown in Figure 7–155, click <Service>.

The Select Service dialog is displayed, as shown in Figure 7–156.
36. Select a File type Business Service and click Submit.
You are returned to the Edit Stage Configuration workspace area, as shown in Figure 7–157.

![Figure 7–157  Edit Stage Configuration Workspace Area](image)

37. In the Request Actions area, click Add an Action, select Reporting from the menu, and click Log.

![Figure 7–158  Edit Stage Configuration Workspace Area](image)

38. As shown in Figure 7–158, click <Expression> to edit the expression.
The XQuery/XSLT Expression Editor is displayed, as shown in Figure 7–159.
39. In the XQuery Text area, type $body.

40. Click Validate and then Save.

As shown in Figure 7–160, you are returned to the Edit Stage Configuration workspace area.

Figure 7–160 Edit Stage Configuration Workspace Area

41. Type any annotation/comments in the text box (for example, **********Response Body**********).

42. Select Error from the severity level list.

43. Click Validate and then Save.

The completed Pipeline is displayed, as shown in Figure 7–161.
44. Click Save.

45. As shown in Figure 7–162, click Activate in the Change Center area to activate your changes in the Oracle Service Bus session.

Figure 7–162  Activate Button

46. As shown in Figure 7–163, copy and paste an input XML file in the input folder you have configured.
Figure 7–163  Input XML File Copied to the Configured Input Folder

An output XML file is received in the destination folder, as shown in Figure 7–164.

Figure 7–164  Output XML File Received in the Destination Folder
This chapter explains the limitations and workarounds when connecting to Siebel. It contains the following topics:

- Section 8.1, "Troubleshooting"
- Section 8.2, "BSE Error Messages"

8.1 Troubleshooting

This topic provides troubleshooting information for Siebel, separated into four categories:

- General Usage Notes for the Oracle Application Adapter for Siebel
- Application Explorer
- Siebel
- Oracle Adapter J2CA
- Oracle Adapter Business Services Engine (BSE)

---

**Note:** Log file information that can be relevant in troubleshooting can be found in the following locations:

- The Oracle Adapter J2CA trace information can be found under the following directory:
  
  `<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\config\config_name\log`

- BSE trace information can be found under the following directory:
  
  `<ADAPTER_HOME>\user_projects\domains\base_domain\servers\soa_server1\stage\ibse\ibse.war\ibselogs`

- The log file for Application Explorer can be found under the following directory:
  
  `<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\tools\iwae\bin`

---

**General Usage Notes for the Oracle Application Adapter for Siebel**

The Oracle Application Adapter for Siebel is subject to the following limitations:

- The HTTPS protocol is not supported for services and events.
- Updates for multi-value (MVG) fields with join specifications are not supported.
- When a connection is lost, the adapter does not automatically reconnect to Siebel.

**Application Explorer**

To use Application Explorer on **Windows** for debugging or testing purposes:

1. Ensure that Oracle WebLogic Server is started, which is where Application Explorer is deployed.
2. Start Application Explorer by clicking the Windows **Start** menu, selecting **All Programs, Oracle Application Adapters**, and clicking **Application Explorer**, as shown in **Figure 8–1**.

**Figure 8–1  Oracle Application Adapters Program Menu**

You can also start Application Explorer by executing the **ae.bat** file, which is located in the following directory:

<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\tools\iwae\bin\ae.bat

It is a good practice to create a shortcut for the **ae.bat** file on your desktop.

If you are using a UNIX or Linux platform, then you can start Application Explorer by executing the **iwae.sh** file.

<table>
<thead>
<tr>
<th>Error</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Siebel does not appear in the Application Explorer Adapter node list.</td>
<td>Ensure that the Siebel jar files supplied with your Siebel distribution media have been placed in the <code>&lt;ADAPTER_HOME&gt;\soa\thirdparty\ApplicationAdapters\lib</code> directory. For example, for Siebel 7.03 environments, the <code>SiebelJI_Common.jar</code> and <code>SiebelJI_enu.jar</code> should be placed in this directory.</td>
</tr>
<tr>
<td>Target Type drop down contains only Java Data Bean Connection and COM connection type is desired.</td>
<td>Ensure that the Siebel thin client is installed correctly on the system hosting Application Explorer so that appropriate COM environment is available.</td>
</tr>
<tr>
<td>An error message that includes the name of the Siebel Gateway server appears when you try to connect to a Siebel target. For example, <strong>Problem activating adapter</strong> (&lt;server_name&gt;). Check logs for more information.</td>
<td>Ensure that the name of the Siebel Gateway server is correctly defined for the target you are using.</td>
</tr>
<tr>
<td>You receive the following error when trying to connect to a Siebel target: <strong>Problem activating adapter. (You have entered an invalid set of logon parameters. Please type in your logon parameters again.). Check logs for more information.</strong></td>
<td>Ensure that the User ID and password parameter values to connect to your Siebel system are correct.</td>
</tr>
</tbody>
</table>
You receive the following error when trying to connect to a Siebel target:
Problem activating adapter. (Couldn't get nameserver connection). Check logs for more information.

You receive the following error when attempting to connect to a Siebel target:
Problem activating adapter. (NSReadKey request failed (no error information)...). Check logs for more information.

You receive the following error when attempting to connect to a Siebel target:

A successful connection is made to Siebel environment but no values are available in Business Object, Business Service, and Integration Object nodes in Application Explorer tree.

Logon failure error at run-time.

The following exception occurs when you start Application Explorer by activating ae.bat (not iaexplorer.exe):
java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.BouncyCastleProvider

Error Solution

Check on network connectivity to Siebel environment. Correct networking problem and retry connection.

Ensure that the values defined for Siebel Server, Enterprise Name, and Object Manager for the target you are using are correct, and retry the connection

Ensure that the value of the Language parameter on the Advanced tab is defined correctly for the target you are using to connect to your Siebel system (for example, enu for English).

The Repository Name specified on the Advanced tab in the Siebel target configuration is either void or empty of any components in the targeted Siebel environment or that Repository Name is not valid for the targeted Siebel environment. Verify that the Repository Name is valid and contains components for interrogation then re-connect.

If the password for connecting to your Siebel system is not specified when creating a target or with the Edit option in Application Explorer, then you are unable to connect to Siebel. The connection password is not saved in repository.xml. Update the password using the Edit option in Application Explorer, then restart the application server.

This is a benign exception. It does not affect adapter functionality. Download BouncyCastle files from:
ftp://ftp.bouncycastle.org/pub
### Error | Solution
--- | ---
Unable to start Application Explorer in a Solaris environment. The following exception is thrown in the console:

```
java.util.ResourceException: IWAFManagedConnectionFactory: License violation.
at com.ibi.afjca.spi.IWAFManagedConnectionFactory.createConnectionFactory(IWAFManagedConnectionFactory.java:98)
at com.iwaysoftware.iwae.common.JCATransport.getConnectionFactory(JCATransport.java:133)
at com.iwaysoftware.iwae.common.JCATransport.initJCA(JCATransport.java:69)
at com.iwaysoftware.iwae.common.JCATransport.<init>(JCATransport.java:62)
at com.iwaysoftware.iwae.common.AdapterClient.<init>(AdapterClient.java:85)
at com.ibi.bse.ConfigWorker.run(ConfigWorker.java:41)
at java.lang.Thread.run(Thread.java:534)
```

Could not create the connection factory.

---

### Siebel

The error messages listed can occur when using the adapter with either a BSE or Oracle Adapter J2CA repository project.

### Error | Solution
--- | ---
A successful connection is made to Siebel environment but no values are available in Business Object, Business Service, and Integration Object nodes in Application Explorer tree.

The Repository Name specified on the Advanced tab in the Siebel Target configuration is either void or empty of any components in the targeted Siebel environment or that Repository Name is not valid for the targeted Siebel environment. Verify that the Repository Name is valid and contains components for interrogation then re-connect.

When executing a request, the following error message appears:

```
AdapterException: Unsupported Action: {0} Tquery
```

Verify that method is available for specific request by verifying schema.

When executing a request, the following error message appears:

```
AdapterException: Field 'NFame' does not exist in definition for business component 'Account'. Please ask your systems administrator to check your application configuration.
```

Ensure that field names are valid within request document by referring to schema for that specific object, and then re-submit the request.
8.2 BSE Error Messages

This topic discusses the different types of errors that can occur when processing Web services through BSE.

8.2.1 General Error Handling in BSE

BSE serves as both a SOAP gateway into the adapter framework and as the engine for some of the adapters. In both design time and run-time, various conditions can cause errors in BSE when Web services that use adapters run. Some of these conditions and resulting errors are exposed the same way, regardless of the specific adapter; others are exposed differently, based on the adapter being used. This topic explains what you can expect when you encounter some of the more common error conditions on an adapter-specific basis.

Usually the SOAP gateway (agent) inside BSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, then how it is exposed depends on the adapter and the API or interfaces that the adapter uses. A few scenarios cause the SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside BSE receives an invalid SOAP request, a SOAP fault element is generated in the SOAP response. The SOAP fault element contains fault string and fault code elements. The fault code contains a description of the SOAP agent error.

The following SOAP response document results when BSE receives an invalid SOAP request:

```xml

When connecting to releases before Siebel 7.7 using the Java Data Bean Interface, you cannot reconnect after initial connection loss. This might occur when Application Explorer experiences a brief loss of network connection or if the Siebel Server or Gateway Service is restarted while Application Explorer is logged into the Siebel application.

The following error may occur when adding a service node for a Business Service that includes methods containing method arguments having hierarchy data types.

If you enter a valid XMLCharEncoding value such as UTF-8 or UTF-16, then the following error is received:

Invocation of Service failed.

Oracle Adapter J2CA

<table>
<thead>
<tr>
<th>Error</th>
<th>Solution</th>
</tr>
</thead>
</table>
| In Application Explorer, the following error message appears when you attempt to connect to an Oracle Adapter J2CA configuration: Could not initialize JCA | In the Details tab in the right pane, ensure that the directory specified in the Home field points to the correct directory, for example:  
<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters \tools\iwae\bin\...\..\..|

Error Solution

In Application Explorer, the following error message appears when you attempt to connect to an Oracle Adapter J2CA configuration:

Could not initialize JCA

Restart Oracle WebLogic Server and Application Explorer to log in successfully to the Siebel application. This is a known Siebel API issue. For more information, see Siebel Alert 984.

The method argument XMLCharEncoding is not supported. Leave this element blank in the XML payload.
In this example, BSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this Web service.

8.2.2 Adapter-Specific Error Handling

When an adapter raises an exception during run-time, the SOAP agent in BSE produces a SOAP fault element in the generated SOAP response. The SOAP fault element contains fault code and fault string elements. The fault string contains the native error description from the adapter target system. Since adapters use the target system interfaces and APIs, whether an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in BSE, and that request is invalid based on the WSDL for that service, then the adapter may raise an exception yielding a SOAP fault.

While it is almost impossible to anticipate every error condition that an adapter may encounter, the following is a description of how adapters handle common error conditions and how they are then exposed to the Web services consumer application.

Oracle Application Adapter for Siebel Invalid SOAP Request

If Oracle Application Adapter for Siebel receives a SOAP request message that does not conform to the WSDL for the Web services being executed, then the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Server</faultcode>
      <faultstring>XD[FAIL] Parse failure (IS) 3: org.xml.sax.SAXParseException: Premature end of file.</faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From Siebel Request

If Oracle Application Adapter for Siebel cannot connect to Siebel when executing a Web service, then the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Server</faultcode>
      <faultstring><Exception> - major:4096 minor: -1 message:NSReadKey request 11 was abandoned after 37846ms connection:12a due to Connection shutdown request Connection reset by peer:JVM_recv in socket input stream stream read DetailedMessage:Unknown</Exception></faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```
Oracle WebLogic Server Integration Adapters

Oracle Adapters connect BSE to adapters whose engines are other Oracle servers. Therefore, since this type of adapter is used to connect BSE to many different target systems, the error handling behavior is consistent. Check the user guide for your adapter to see if you require the Oracle WebLogic Server Integration Adapter when running Web services.

Invalid SOAP Request

If Oracle Application Adapter for Siebel receives a SOAP request message that does not conform to the WSDL for the Web services being executed, then the following SOAP response is generated.

```xml
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Body>
<SOAP-ENV:Fault>
<faultcode>SOAP-ENV:Server</faultcode>
<faultstring>RPC server connection failed: Connection refused: connect</faultstring>
</SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From Oracle WebLogic Server Adapter Request

If Oracle Application Adapter for Siebel executes a SOAP request using input parameters passed that do not match records in the target system, then the following SOAP response is generated.

```xml
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
<SOAP-ENV:Body>
<m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
xmlns="urn:schemas-iwaysoftware-com:iwse"
cid="2A3CB42703EB20203F91951B89F3C5AF">
<RunDBQueryResult run="1" />
</m:RunDBQueryResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

**Note:** The condition for this adapter does not yield a SOAP fault.
This chapter describes advanced user tools. It contains the following topics:

- Section 9.1, "Web Services Policy-Based Security"
- Section 9.2, "Migrating Repositories"

9.1 Web Services Policy-Based Security

Application Explorer provides a security model called Web services policy-based security. The following topics describe how the feature works and how to configure it.

Web services provide a layer of abstraction between the back-end business logic and the user or application running the Web service. Easy application integration is enabled. However, the issue of controlling the use and implementation of critical and sensitive business logic that is run as a Web service is raised.

Application Explorer controls the use of Web services that use adapters, using a feature called policy-based security. This feature enables an administrator to apply "policies" to Business Services (Web services) to deny or permit their execution.

A policy is a set of privileges dealing with the execution of a Business Service (iBS) that can be applied to an existing or new iBS. When you set specific rights or privileges inside a policy, you do not have to re-create privileges for every iBS that has security concerns in common with other Business Services. Instead, you reuse a policy on multiple Business Services.

The goal of the feature is to secure requests at both the transport and the SOAP request level transmitted on the wire. Some of the policies do not deal with security issues directly, but do effect the run-time behavior of the Web services to which they have been applied.

The iBS administrator creates an “instance” of a policy type, names it, associates individual users or groups (a collection of users), and then applies that policy to one or more Business Services.

You can assign a policy to an iBS, or to a method within an iBS. If a policy is only applied to a method, then other methods in that iBS is not governed by it. However, if a policy is applied to the iBS, all methods are governed by it. At run-time, the user ID and password that are sent to BSE in the SOAP request message are checked against the list of users for all policies applied to that specific iBS. The policy type that is supported is Resource Execution, which dictates who can or cannot execute the iBS.

When a policy is not applied, the default value for an iBS is to "grant all". For example, anybody can execute the iBS, until the Resource Execution policy is associated to the iBS. At that time, only those granted execution permissions, or users not part of the group that has been denied execution permissions, have access to the iBS.
9.1.1 Configuring Web Services Policy-Based Security

The following procedures describe how to configure Web services policy-based security.

Creating and Associating a User with a Policy

Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Application Explorer.

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. For information on creating a new configuration, see Chapter 2, "Configuring Oracle Application Server Adapter for Siebel".
3. Select Connect.

   As shown in Figure 9–1, nodes appear for Adapters and Business Services (also known as Web services).

![Figure 9–1 Siebel Configuration, SampleConfig](image)

Perform the following steps:

a. Expand the Business Services node.
b. Expand the Configuration node.
c. Expand the Security node.
d. Expand the Users and Groups node.

4. As shown in Figure 9–2, right-click Users and then New User.

![Figure 9–2 New User Option](image)

The New User dialog box is displayed, as shown in Figure 9–3.
Perform the following steps:

a. In the **Name** field, enter a user ID.

b. In the **Password** field, enter the password associated with the user ID.

c. In the **Description** field, enter a description of the user (optional).

5. Click **OK**.

   The new user is added under the Users node, as shown in **Figure 9–4**.

### Figure 9–4  New User Added Under the Users Node

![Users](image)

#### Creating a Group to Use With a Policy

To create a group to use with a policy:

1. Start Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. For information on creating a new configuration, see Chapter 2, “Configuring Oracle Application Server Adapter for Siebel”.

3. Select **Connect**.

   As shown in **Figure 9–5**, nodes appear for **Adapters** and **Business Services** (also known as Web services).

#### Figure 9–5  Adapters and Business Services Nodes

![Adapters](image)

Perform the following steps:

a. Expand the **Business Services** node.

b. Expand the **Configurations** node.

c. Expand the **Security** node.

d. Expand the **Users and Groups** node.

4. Right-click **Groups** and select **New Group**, as shown in **Figure 9–6**.
The New Group dialog box is displayed, as shown in Figure 9–7.

Perform the following steps:

a. In the Name field, enter a name for the group.

b. In the Description field, enter a description for the group (optional).

c. From the available list of users in the left pane, select one or more users and add them to the Selected list by clicking the double right facing arrow.

5. When you have selected at least one user, click **OK**.

As shown in Figure 9–8, the new group is added under the Group node.
Creating an Execution Policy

An execution policy governs who can execute the Business Services to which the policy is applied.

To create an execution policy:

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. For information on creating a new configuration, see Chapter 2, “Configuring Oracle Application Server Adapter for Siebel”.
3. Select Connect.

As shown in Figure 9–9, nodes appear for Adapters and Business Services (also known as Web services).

![Figure 9–9 Adapters and Business Services Nodes](image)

Perform the following steps:

a. Expand the Business Services node.

b. Expand the Configurations node.

c. Expand the Security node.

d. Expand the Policies node.

4. As shown in Figure 9–10, right-click Policies and select New Policy.

![Figure 9–10 New Policy Option](image)

The New policy dialog box is displayed, as shown in Figure 9–11.
Perform the following steps:

a. In the **Name** field, enter a name for the policy.

b. From the **Type** list, select **Execution**.

c. In the **Description** field, enter a description for the policy (optional).

d. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right facing arrow.

---

**Note:** This user ID is checked against the value in the user ID element of the SOAP header sent to BSE in a SOAP request.

---

5. When you have selected at least one user, click **OK**.

6. Click **Next**.

   The New Policy permissions dialog box is displayed.

7. When you have selected at least one user, click **OK**.

8. Click **Next**.

   The New Policy permissions dialog box is displayed, as shown in Figure 9–12.
To grant permission to a user or group to execute an iBS, select the user or group and move them into the Execution Granted list by selecting the double left facing arrow.

To deny permission to a user or group to execute an iBS, select the user or group and move them into the Execution Denied list by selecting the double right facing arrow.

9. Click OK.

The following pane summarizes your configuration, as shown in Figure 9–13.

Using the IP and Domain Restrictions Policy Type

You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to BSE and therefore need not be applied to individual Web services. You need not create a policy, however, you must enable the Security Policy option in Application Explorer.

1. Start Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. For information on creating a new configuration, see Chapter 2, "Configuring Oracle Application Server Adapter for Siebel".

3. Select Connect.
As shown in Figure 9–14, nodes appear for Adapters and Business Services (also known as Web services).

**Figure 9–14  Adapters and Business Services Nodes**

Perform the following steps:

a. Expand the Business Services node.

b. Expand the Configurations node.

c. Expand the Security node.

4. As shown in Figure 9–15, right-click IP and Domain and select New IP and Domain Restriction.

**Figure 9–15  New IP and Domain Restriction Option**

The New IP and Domain Restriction dialog box is displayed, as shown in Figure 9–16.

**Figure 9–16  New IP and Domain Restriction Dialog**

Perform the following steps:

a. In the IP(Mask)/Domain field, enter the IP or domain name using the following guidelines:

   If you select Single (Computer) from the Type list, then you must provide the IP address for that computer. If you only know the DNS name for the computer, then click DNS Lookup to obtain the IP Address based on the DNS name.

   If you select Group (of Computers), then you must provide the IP address and subnet mask for the computer group.
If you select **Domain**, then you must provide the domain name.

b. From the **Type** list, select the type of restriction.

c. In the **Description** field, enter a description (optional).

d. To grant access, select the **Grant Access** check box.

5. Click OK.

The new domain is added under the IP and Domain node.

The following pane summarizes your configuration, as shown in Figure 9–17.

**Figure 9–17  Summary of IP and Domain Configuration**

- **IP Address (Mask) / Domain**: www.yahoo.com  
- **Type**: Domain  
- **Access**: Denied  
- **Description**

### 9.2 Migrating Repositories

During design time, the Oracle repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at run-time. For management purposes, you can migrate BSE and J2CA repositories that are configured for Oracle to new destinations without affecting your existing configuration. For example, you may want to migrate a repository from a test environment to a production environment.

**Migrating a BSE Repository**

To migrate a BSE repository:

1. Copy the BSE control service URL, for example:
   
   `http://localhost:8001/ibse/IBSEServlet/admin/iwcontrol.ibs`

2. Open a third party XML editor, for example, XMLSPY.

3. From the menu bar, click **SOAP**.

   A list of options appears, as shown in Figure 9–18.

**Figure 9–18  Create new SOAP request Option**

```
<table>
<thead>
<tr>
<th>SOAP</th>
<th>Tools</th>
<th>Window</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create new SOAP request</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Send request to server</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change SOAP request parameters</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

4. Select **Create new SOAP request**.

   The WSDL file location dialog box is displayed, as shown in Figure 9–19.
Perform the following steps:

**a.** In the **Choose a file** field, paste the BSE control service URL.

**b.** Append `?wsdl` to the URL, for example:

```
http://localhost:8001/ibse/IBSEServlet/admin/iwcontrol.ibs?wsdl
```

5. Click **OK**.

The soap operation name dialog box displays the available control methods, as shown in **Figure 9–20**.

6. Select the **MIGRATEREPO(MIGRATEREPO parameters)** control method and click **OK**.

---

**Note:** The MIGRATEREPO(MIGRATEREPO parameters) control method is available from the BSE administration console. This control method migrates all Web services to the new (empty) repository. You can choose to migrate select Web services only.

The following window is displayed, showing the structure of the SOAP envelope, as shown in **Figure 9–21**.
7. Locate the **Text view** icon in the toolbar, as shown in Figure 9–22.

**Figure 9–22  Text View Icon**

8. To display the structure of the SOAP envelope as text, click the **Text view** icon. The `<SOAP-ENV:Header>` tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```xml
<m:MIGRATEREPO xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config" version="">
  <m:repositorysetting>
    <m:rname>oracle</m:rname>
    <m:rconn>String</m:rconn>
    <m:rdriver>String</m:rdriver>
    <m:ruser>String</m:ruser>
    <m:rpwd>String</m:rpwd>
  </m:repositorysetting>
  <m:servicename>String</m:servicename>
</m:MIGRATEREPO>
```

Perform the following steps:

a. For the `<m:rconn>` tag, replace the String placeholder with a repository URL where you want to migrate your existing BSE repository. The Oracle repository URL has the following format:

   `jdbc:oracle:thin:@[host]:[port]:[sid]`

b. For the `<m:rdriver>` tag, replace the String placeholder with the location of your Oracle driver.
Migrating Repositories

9-12

10. Perform one of the following migration options.

- If you want to migrate a single Web service from the current BSE repository, then enter the Web service name in the `<m:servicename>` tag, for example:
  
  `<m:servicename>SiebelService1</m:servicename>`

- If you want to migrate multiple Web services from the current BSE repository, then duplicate the `<m:servicename>` tag for each Web service, for example:
  
  `<m:servicename>SiebelService1</m:servicename>`
  `<m:servicename>SiebelService2</m:servicename>`

- If you want to migrate all Web services from the current BSE repository, then remove the `<m:servicename>` tag.

11. From the menu bar, click SOAP and select Send request to server, as shown in Figure 9–23.

*Figure 9–23  Send request to server Option*

<table>
<thead>
<tr>
<th>SOAP</th>
<th>Tools</th>
<th>Window</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create new SOAP request</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Send request to server</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change SOAP request parameters</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Your BSE repository and any Web services you selected are now migrated to the new Oracle repository URL you specified.

### Migrating a J2CA Repository

To migrate a J2CA repository:

1. Navigate to the location of your J2CA configuration directory where the repository schemas and other information is stored, for example:

   `<ADAPTER_HOME>\soa\thirdparty\ApplicationAdapters\config\JCA_CONFIG`

   Where JCA_CONFIG is the name of your J2CA configuration.

2. Locate and copy the `repository.xml` file.

3. Place this file in a new J2CA configuration directory to migrate the existing repository.

   Your J2CA repository is migrated to the new J2CA configuration directory.
Using Siebel Workflows

When using Siebel XML to integrate with Siebel Integration Objects, the interface uses a Siebel Workflow.

Note: This appendix is intended as a supplement to the documentation designed for Oracle Application Adapter for Siebel user and is not intended as a substitute for Siebel documentation. For complete and up-to-date information on Siebel Workflow and policy topics, see the Siebel Bookshelf for your Siebel system.

This appendix contains the following topics:

- Section A.1, "Overview"
- Section A.2, "Creating a Siebel Workflow"

A.1 Overview

A Siebel Workflow is defined within Siebel to emit or to receive Siebel XML. In either case, emitting or receiving is handled by Siebel transport services for MQSeries, File, or HTTP. The following topics describe the use and creation of workflows that employ the supported transport services.

A.1.1 Siebel Workflows

A Siebel Workflow is a series of Siebel Business Services linked together to accomplish a business task. You create workflows using the Siebel Client Workflow Administration screens. Workflows are invoked through one of the following methods:

- Using a workflow policy
- Using a run-time event (Siebel Event)
- Using a script (eScript or Siebel VB)

The following topic briefly describes how to invoke the workflow through a policy condition.

See Also:

Siebel Bookshelf documentation for more information on policy and other methods.
A.1.2 Using a Policy to Invoke a Siebel EAI Workflow

A workflow policy is defined by a set of conditions that performs a set of defined actions. A Siebel workflow policy consists of:

- Conditions that define circumstances, based on changes in the state of a Siebel database.
- Actions that define steps taken when conditions are fulfilled.

Creating a policy to invoke a workflow as an action involves the following steps:

1. Define an action to be executed after a policy is triggered. Use the Run Integration Process program.
2. Create a policy by setting conditions and selecting appropriate policy groups and actions.
3. Activate the policy by choosing an activation date.
4. Run the Generate Triggers server task from Server Administration windows to set the conditions to be monitored.
5. Start the Workflow Monitor agent after editing with the appropriate policy group (to which your policy belongs) to evaluate whether to perform an action.
6. Start the Workflow Action Agent server task from Server Administration windows to perform the action.

A.1.3 Siebel Workflow - Outbound

When a Siebel Workflow is triggered based on a Siebel policy, run-time, or script (eScript or Siebel VB) event, the result is the generation of a Siebel XML document that is placed on one of the Siebel transports. For example, when you add a new account in the Siebel Call Center application, you can design and configure a workflow to be triggered on the account transaction. You can design the workflow to extract the data for the new record, convert it to Siebel XML, and then, place it on an MQSeries message queue.

In this example, the Siebel Workflow process executes the following series of Siebel Business Services:

1. Calls the Siebel EAI Siebel Adapter that queries for the newly updated account record and places the data in its original internal structure into memory.
2. Calls the Siebel EAI XML Converter that converts the data into an XML message.
3. Calls the Siebel EAI MQSeries Transport that places the newly created XML message into the appropriate MQSeries message queue.

After the message is placed in the message queue, it is retrieved by Oracle Application Adapter for Siebel 6.3 and higher. The following Workflow sequence illustrates the previous steps, as shown in Figure A–1.
A.1.4 Siebel Workflow - Inbound

A Siebel Workflow that is triggered by an external event begins by receiving a Siebel XML document placed on one of its transports. The result might be the update of a Siebel record using the XML as input, for example, when a new account is added in another CRM system but also must be updated in the Siebel Call Center application. You can design and configure a Workflow to receive or listen on an MQSeries message queue. Upon receipt of the XML message, the Workflow processes the transaction into the Siebel system to update the record.

In this example, upon receipt of the Siebel XML message in the message queue, the Siebel MQSeries Receiver server task initiates a Siebel Workflow process, which in turn executes a series of Siebel Business Services as follows:

1. Calls the Siebel EAI XML Converter, which converts the XML message into Siebel internal format.
2. Calls the Siebel EAI Siebel Adapter, which applies the newly updated account record based on the methods defined in its service.

The following is a sample of the Workflow process, as shown in Figure A–2.

A.2 Creating a Siebel Workflow

The following topics include procedures for creating Siebel Workflows in the Siebel Workflow Administration window.

A.2.1 Creating a Siebel Workflow for an Event Using MQSeries Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the MQSeries transport.

The following is a Siebel Workflow Administration window, as shown in Figure A–3.
The following procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application. The Workflow is then placed on an MQSeries message queue.

To create a Siebel Workflow:

1. In the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties.

   The Account message contains Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the workflow has converted to XML.

   The following window is displayed, showing the Process Properties tab active, as shown in Figure A–4.
2. Use the Siebel Workflow Administration windows to create a Workflow.


   The Business Service obtains the Account information from Siebel using the Query method.

   Output from this Business Service is generated in hierarchical format, as shown in Figure A–5.
4. Define an EAI XML Converter Business Service step and call it `Convert to XML`.

It is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format, as shown in Figure A–6.
5. Define an EAI MQSeries server transport Business Service step and call it Send to Q, as shown in Figure A–7.

   It is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to MQSeries using the Send method.
A.2.2 Creating a Siebel Workflow for an Event Using File Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the File transport.

The following window is displayed with the Process Designer tab active, as shown in Figure A–8.
This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

To create a Siebel Workflow:
1. On the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties, as shown in Figure A–9. Account message contains the Siebel Account data in hierarchical format.
Account XML specifies which Siebel Account data the Workflow converted to XML.

2. Use the Siebel Workflow Administration windows to create a Workflow. As shown in Figure A–10, the following is an example of a Siebel Workflow Administration window.

The Business Service obtains the Account information from Siebel using the Query method.

Output from this Business Service is generated in hierarchical format.
4. As shown in Figure A–11, define an EAI XML Converter Business Service step and call it Convert Account Data to XML.

   This Business Service is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format.
5. As shown in Figure A–12, define an EAI File Transport Business Service step and call it Send Account Data. This Business Service is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to the file system in a specified directory using the Send method.

A.2.3 Creating a Siebel Workflow for an Event Using HTTP Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for exporting Siebel Account record information using the HTTP transport. This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application.

To create a Siebel Workflow:
1. As shown in Figure A–13, in the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties. Account message contains the Siebel Account data in hierarchical format. Account XML specifies the Siebel Account data that the Workflow has converted to XML.

2. Use the Siebel Workflow Administration windows to create a Workflow.
3. As shown in Figure A–14, define an EAI Siebel Adapter Business Service step to receive an instance of Account data and call it Get New Account.

The Business Service obtains the Account information from Siebel using the Query method.

Output from this Business Service is generated in hierarchical format.

4. Define an EAI XML Converter Business Service step and call it Convert to XML.

This Business Service is defined to receive the Account data from the EAI Siebel Adapter Business Service in hierarchical format and convert it to XML format.
5. Define an EAI HTTP Transport Business Service step and call it `Send HTTP`, as shown in Figure A–15.

This Business Service is defined to receive the Account data from the EAI XML Converter Business Service in Siebel XML format and send the Account XML to HTTP using the Send method.

### A.2.4 Creating a Siebel Workflow for a Service Using MQSeries Transport

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The Workflow was designed for importing Siebel Account record information through the MQSeries Transport.

Figure A–16 shows a sample Siebel Workflow Administration window.
This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application.

To create a Siebel Workflow:
1. In the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties, as shown in Figure A–17.

   Account message contains the Siebel Account data in hierarchical format.

   Account XML specifies the Siebel Account data that the Workflow converted to XML.
2. Define an EAI MQSeries Server Transport Business Service step and call it Receive, as shown in Figure A–18.

The Business Service is defined to receive the Account data from the MQSeries message queue.

The EAI MQSeries Server Transport Business Service receives the Account data in Siebel XML format and sends it to the EAI XML Converter Business Service.
3. Define an EAI XML Converter Business Service step and call it Get XML from MQ & Convert to XML, as shown in Figure A–19.

This Business Service is defined to receive the Account data from the EAI MQSeries Server Transport Business Service in XML format and convert it to hierarchical format.
4. Define an EAI Siebel Adapter Business Service step and call it **Update Account**, as shown in **Figure A–20**.

   This Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format.

   The Business Service applies the Account information into Siebel using the Insert or Update method.

**A.2.5 Creating a Siebel Workflow for a Service Using File Transport**

The following procedure is an example of a Siebel Workflow illustrated in the Siebel Workflow Administration window. The workflow was designed for importing Siebel Account record information through the File transport.

This procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

The following is a Siebel Workflow Administration window with the Process Designer tab active, as shown in **Figure A–21**.
To create a Siebel Workflow:

**Figure A–22 Process Properties Tab of the Workflow Process Window**
1. In the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties, as shown in Figure A–22. Account message contains the Siebel Account data in hierarchical format.

Account XML specifies the Siebel Account data that the workflow converted to XML.

Figure A–22  EAI File Transport Business Service Step

2. Define an EAI FileTransport Business Service step and call it Receive Account Data, as shown in Figure A–23.

The Business Service is defined to receive the Account data from the file system. The EAI File Transport Business Service receives the Account data in Siebel XML format and sends it to the EAI XML Converter Business Service.
3. Define an EAI XML Converter Business Service step and call it `Convert from XML`, as shown in Figure A–24.

This Business Service is defined to receive the Account data from the EAI File Transport Business Service in XML format and convert it to hierarchical format.
4. Define an EAI Siebel Adapter Business Service step and call it Update or Insert New Account, as shown in Figure A–25.

This Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format.

The Business Service applies the Account information into Siebel using the Insert or Update method.

**A.2.6 Creating a Siebel Workflow for a Service Using HTTP Transport**

The following procedure is an example of a Siebel workflow illustrated in the Siebel Workflow Administration window, as shown in Figure A–26. The Workflow was designed for importing Siebel Account record information through the HTTP transport, as shown in .
The following procedure describes how to create a Siebel Workflow that generates Siebel XML when an Account record is updated in the Siebel Call Center application and then places Siebel XML on the file system.

To create a Siebel Workflow:
1. In the Process Properties tab of the Workflow Process window, define the Account message and Account XML process properties, as shown in Figure A–27.

Account message contains the Siebel Account data in hierarchical format.

Account XML specifies the Siebel Account data that the workflow converted to XML.
2. Define an EAI XML Converter Business Service step and call it `XML to Property Set`, as shown in Figure A–28.

The Business Service is defined to receive the Account data from the EAI HTTP Transport Business Service in XML format and convert it to hierarchical format.
3. Define an EAI Siebel Adapter Business Service step and call it **Update Siebel**, as shown in Figure A–29.

The Business Service is defined to receive from the EAI XML Converter Business Service the instance of Account data in hierarchical format. The Business Service applies the Account information into Siebel using the Insert or Update method.
adapter
Provides universal connectivity by enabling an electronic interface to be accommodated (without loss of function) to another electronic interface.

agent
Supports service protocols in listeners and documents.

business service
Also known as a Web service. A Web service is a self-contained, modularized function that can be published and accessed across a network using open standards. It is the implementation of an interface by a component and is an executable entity.

channel
Represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by an adapter.

listener
A component that accepts requests from client applications.

port
 Associates a particular business object exposed by the adapter with a particular disposition. A disposition is a URL that defines the protocol and location of the event data. The port defines the end point of the event consumption.
access rights, 9-1
adapter configuration
  updating, 3-2
adapter exceptions, 8-4
adapter types
  resource, 3-2
adapters, 1-1
  configuring, 3-2
  deploying, 3-1 to 3-2, 4-1
  integrating with BPEL Process Manager, 4-1, 6-1
  troubleshooting, 8-1 to 8-7
Adapters node, 4-47, 4-49
Add Channel dialog box, 2-30, 4-44
Add Target dialog box, 2-7
Advanced tab, 2-8, 2-34, 2-36
ae batch script, 8-2
application adapters
  configuring, 3-2
  deploying, 3-1 to 3-2, 4-1
  integrating with BPEL Process Manager, 4-1, 6-1
  troubleshooting, 8-1 to 8-7
Application Explorer, 1-2, 1-5, 2-11 to 2-13, 2-24
debugging and, 8-2 to 8-3
J2CA configuration and, 3-2
OracleAS Adapter J2CA and, 8-5
security and, 9-1, 9-7
targets and, 4-49
testing and, 8-2 to 8-3
troubleshooting, 8-2 to 8-3
WSDL files and, 4-44
application system objects
  viewing, 2-11
Available list, 9-4, 9-6

B
BPEL Designer, 4-1, 6-1
BPEL Process Manager
  integrating with adapters, 4-1, 6-1
OracleAS Adapter for Siebel and, 4-1
browsing metadata, 2-10
BSE (OracleAS Adapter Business Services Engine)
  configuring, 2-3
  connection access to, 9-7
  troubleshooting, 8-5
BSE control service URL, 9-9
BSE repositories
  migrating, 9-9
BSE URL field, 2-4
Business Components, 1-2 to 1-5, 2-13, 2-24
business events, 1-1
Business Integration Manager, 1-5
Business Objects, 1-4, 2-24
  browsing, 2-10
Business Service list, 2-19
business services, 1-2 to 1-5, 2-13, A-1 to A-3
  browsing, 2-10
  creating, 2-24 to 2-25
  deploying, 9-1
testing, 2-26 to 2-27
Business Services node, 9-2 to 9-5, 9-8

C
CCI (Common Client Interface), 3-1
CCI calls, 3-1
channel configuration parameters
  Document type XML, 2-33
  Error Directory, 2-34, 2-36
  File Mask, 2-35
  Message wait interval, 2-34
  Mode of operation, 2-34
  MQ server channel, 2-33
  MQ server host, 2-33
  MQ server port, 2-33
  Poll interval, 2-36
  Polling Location, 2-35
  Port, 2-31, 4-46
  Processing Mode, 2-36
  Queue manager name, 2-33
  Request queue name, 2-34
  Response/Ack Directory, 2-36
  Server port, 2-31, 4-46
  Synchronization Type, 2-31, 2-34, 2-36, 4-46
  Thread limit, 2-34, 2-36
channel field, 4-50
channel types
  File, 2-35
  HTTP, 2-30, 4-44 to 4-47
  MQSeries, 2-33
Group (of Computers) option, 9-8
groups
  creating, 9-3
Groups node, 9-4

H
Hostname parameter, 2-4
HTTP channel, 2-30, 4-44 to 4-47
HTTP transport, A-13, A-25

I
inbound J2CA services, 4-44
inbound processing, 1-1 to 1-5, A-3
input arguments, 4-70, 6-93
Input Arguments tab, 4-70, 6-93
installation directories, 3-4
instances of policy types, 9-1
integration access methods
  OracleAS Adapter for Siebel and, 1-5
Integration Object (IO) node, 2-20
  creating, 4-48
Integration Objects, 1-2, 2-17
  browsing, 2-10
IO (Integration Object) node, 2-20
  creating, 4-48
IP (Mask)/Domain field, 9-8
IP addresses, 9-8
IP and Domain Restriction policy type, 9-7
IWayConfig parameter, 3-4
IWayHome parameter, 3-4
IWayRepoPassword parameter, 3-4
IWayRepoURL parameter, 3-4
IWayRepoUser parameter, 3-4

J
J2CA (Enterprise Connector for J2EE Connector
  Architecture), 1-1
J2CA architecture
  Oracle Application Server Adapter and, 3-1
J2CA configuration
  Application Explorer and, 3-2
J2CA repositories
  migrating, 9-12
J2CA resource adapters, 1-1, 3-1
J2CA services, 4-44
JAR files, 8-2
Java Data Bean, 1-5
Java program clients, 3-1
JDeveloper, 4-7, 6-1

L
Language parameter, 2-8, 8-3
License and Method dialog box, 2-25
License field, 2-25
licenses, 2-25
list of nodes, 8-2
listeners. See also channels
log files, 8-1
log levels
  overwriting, 3-4
Loglevel parameter, 3-4
logon parameters, 8-2

M
managed connector factories, 3-2
managed connector factory objects, 3-2
  multiple, 3-4
ManagedConnectionFactory parameter, 3-4
Mediator Inbound Process, 5-28
Mediator Outbound Process, 5-2
message interactions
  asynchronous, 1-2
  synchronous, 1-2
message types
  event, 4-44
Message wait interval parameter, 2-34
messages, 1-1
metadata
  browsing, 2-10
  viewing, 2-11
metadata tables, 2-12
Method Name field, 2-25
methods, 9-1
Methods node, 2-26
migrating repositories, 9-9
Mode of operation parameter, 2-34
MQ server channel parameter, 2-33
MQ server host parameter, 2-33
MQ server port parameter, 2-33
MQSeries channel, 2-33
MQSeries transport, A-3, A-16

N
Name field, 9-3 to 9-4, 9-6
navigation paths, 1-4
New Configuration dialog box, 2-3 to 2-5
New Group dialog box, 9-4
New Policy permissions dialog box, 9-6
New User dialog box, 9-2
Node list, 8-2
Node name field, 4-48
nodes
  Adapters, 4-47, 4-49
  Business Services, 9-2, 9-8
  Channels, 2-30, 4-44
  Configuration under Business Services, 9-2 to 9-5, 9-8
Configurations, 2-4, 9-2
  connected, 2-10, 4-47
  disconnected, 2-10
Events, 2-30
Groups, 9-4
Integration Object, 2-20, 4-48
Methods, 2-26
Policies, 9-5
Ports, 4-44
Sample Account, 2-20, 4-48
Security, 9-2 to 9-5, 9-8
Services, 2-26
Siebel, 4-47 to 4-49
Users and Groups, 9-2 to 9-3

O
Object Manager, 1-2
Object Manager parameter, 2-8
Oracle Application Server
deployment of Connector to, 3-1
Oracle Application Server Adapter
J2CA architecture and, 3-1
troubleshooting, 8-7
Oracle databases, 3-4
Oracle JDeveloper, 4-7, 6-1
Oracle repositories
migrating, 9-9
Oracle’s Unified Method (OUM), viii
OracleAS Adapter
installation directory and, 3-4
OracleAS Adapter Application Explorer. See Application Explorer
OracleAS Adapter Business Services Engine (BSE), 1-1, 2-3
troubleshooting, 8-5
OracleAS Adapter for Siebel
BPEL Process Manager and, 4-1
deploying, 1-1, 4-1
integrating with Siebel, 4-8
integration access methods and, 1-5
troubleshooting, 8-1 to 8-7
OracleAS Adapter J2CA, 2-3 to 2-4
Application Explorer and, 8-5
connecting to, 8-5
outbound processing, 1-1 to 1-5, A-2

P
parameter types
channel configuration, 2-31 to 2-36, 4-46
configuration, 3-4
connection, 2-8, 8-2
repository migration, 9-10
Password parameter, 8-2, 9-3
passwords, 3-4
permissions, 9-1
denyng, 9-7
granting, 9-7
policies, 9-1
applying, 9-1
creating, 9-5
Policies node, 9-5
policy types, 9-1
IP and Domain Restriction, 9-7
policy-based security, 9-1 to 9-9
Poll interval parameter, 2-36
Polling Location parameter, 2-35
Port Number parameter, 2-4
Port parameter, 2-31, 4-46
ports
creating, 2-29, 4-49 to 4-50
Ports node, 4-44
privileges, 9-1
setting, 9-1
Process Designer tab, 4-70, 6-93
Process Manager. See BPEL Process Manager
Process Simulator tab, 4-71, 6-94
Processing Mode parameter, 2-36
projects
repository, 2-3 to 2-5
properties, 3-4
Protocol list, 2-20, 4-45, 4-48

Q
Query method, A-11
Queue manager name parameter, 2-33

R
repositories
migrating, 9-9
Repository Manager parameter, 2-8
repository migration parameters, 9-10
Repository Name parameter, 8-3 to 8-4
repository projects
creating, 2-3 to 2-5
Web services and, 2-3
repository URLs, 9-11 to 9-12
repository.xml file, 9-12
REQUEST option, 2-34
Request queue name parameter, 2-34
request schemas, 2-13
Request tab, 2-33, 2-35
REQUEST_ACK option, 2-34
REQUEST_RESPONSE option, 2-34
requests
executing, 8-4
Resource Execution policy type, 9-1
response schemas, 2-13
Response tab, 2-34 to 2-36
Response/Ack Directory parameter, 2-36
runtime, 2-3, 4-8, 4-47 to 4-50
runtime events (Siebel Events), A-1
runtime integration, 4-67, 6-90

S
Sample Account node, 2-20, 4-48
schema considerations, 2-13
Schema location field, 4-48
schemas
creating, 2-13 to 2-15, 2-16
scripts, A-1
security, 9-1 to 9-9
Application Explorer and, 9-1, 9-7
configuring, 9-2
Security node, 9-2 to 9-5, 9-8
security policies
applying, 9-1
creating, 9-5
Security Policy option, 9-7
Selected list, 9-4, 9-6
Server port parameter, 2-31, 4-46
service integration, 4-8
service names, 2-25
Service Provider list, 2-4 to 2-5
service schemas
creating, 2-13 to 2-15
Service-Oriented Architecture (SOA), 4-1
services, 1-1
creating, 2-24 to 2-25
deploying, 9-1
testing, 2-26 to 2-27
workflows and, A-16 to A-29
Services node, 2-26
servlets, 3-1
Siebel
connecting to, 1-4, 2-6 to 2-10, 4-47, 8-2 to 8-4, 8-6
disconnecting from, 2-10
Siebel Business Components, 1-2 to 1-5, 2-13, 2-24
Siebel Business Objects, 1-4, 2-24
browsing, 2-10
Siebel business processes
invoking, 1-4
Siebel Business Services, 1-2 to 1-5, 2-13, A-1 to A-3
browsing, 2-10
Siebel Call Center
starting, 4-67, 6-90
Siebel Client Workflow Administration screens, A-1
Siebel COM Data Interface, 1-5
Siebel Events (run-time events), A-1
Siebel Gateway server, 8-2
Siebel Integration Objects, 1-2, 2-17
browsing, 2-10
Siebel Java Data Bean, 1-5
Siebel node, 4-47 to 4-49
Siebel Object Manager, 1-2
Siebel Tools Schema Wizard, 2-16, 2-19
Siebel Tools window, 2-18
Siebel transports, 1-5
Siebel VB script, A-1
Siebel Workflow Administration window, 4-68, 6-91
Siebel workflow policies, A-2
Siebel Workflows, 1-5, A-1 to A-29
creating, A-3
SiebelJI_Common.jar file, 8-2
SiebelJI_enu.jar file, 8-2
Single (Computer) option, 9-8
SOA (Service-Oriented Architecture), 4-1
SOAP agents, 8-5 to 8-6
SOAP envelopes, 9-10
SOAP faults, 8-5 to 8-6
soap operation name dialog box, 9-10
SOAP requests, 8-5 to 8-7, 9-1, 9-6
creating, 9-9
test, 8-5 to 8-7
errors and, 8-5 to 8-7
SOAP responses, 8-5 to 8-7
starting the Siebel Call Center, 4-67, 6-90
Synchronization Type parameter, 2-31, 2-34 to 2-36, 4-46
system objects
viewing, 2-11
T

target systems
errors and, 8-6
Target Type list, 2-7
targets
connecting to, 2-6 to 2-10, 4-47, 4-49, 8-2 to 8-4
defining, 2-7
deleting, 2-11
disconnecting from, 2-10
deploying, 2-11
testing Web services, 2-26 to 2-27
thin clients, 8-2
Thread limit parameter, 2-34, 2-36
trace information, 8-1
transaction processing, 1-1
transports, 1-5
File, A-8 to A-13, A-21
HTTP, A-13, A-25
MQSeries, A-3, A-16
triggering an event, 4-67 to 4-71, 6-90 to 6-94
troubleshooting, 8-1 to 8-7
Application Explorer, 8-2 to 8-3
BSE, 8-5
Web services, 8-5 to 8-7
Type list, 9-6, 9-8
U

updating adapter configuration, 3-2
UserName parameter, 8-2
users
associating, 9-2
Users and Groups node, 9-2 to 9-3
Users node, 9-3
V

viewing system objects, 2-11
W

Web Service Definition Language (WSDL), 2-28 to 2-29
Web service names, 2-25
Web services
creating, 2-24 to 2-25
deploying, 9-1
integrating, 4-1
repository projects and, 2-3
test, 2-26 to 2-27
troubleshooting, 8-5 to 8-7
Web services policy-based security, 9-1 to 9-9
workflow elements, 4-70, 6-93
workflow policies, A-2
Workflow Process window, A-4 to A-23, A-27
Workflow Processes window, 4-68, 6-91
Workflows, A-1 to A-29
creating, A-3
WSDL (Web Service Definition Language), 2-28 to 2-29
WSDL documents, 4-1
WSDL file location dialog box, 9-9
WSDL File Name field, 4-50
WSDL files, 4-1, 6-1
  Application Explorer and, 4-44
  creating, 4-44

X

XDR schemas
  creating, 2-16

XML documents, 1-4, A-2
XML messages, 1-1, A-2 to A-3
XML schemas, 1-2, 1-5
  creating, 2-13 to 2-15, 2-16