SUN SEEBEYOND

# eWAY™ ADAPTER FOR SIEBEL EAI USER'S GUIDE

**Release 5.1.2**

Part Number: 819-7392-10

Version 20061012140531

# Contents

Chapter 1

# Introducing the Siebel EAI eWay

Welcome to the *Sun SeeBeyond eWay™ Adapter for Siebel EAI User's Guide*. This document includes information about installing, configuring, and using the Sun Java Composite Application Platform Suite Siebel EAI eWay™ Adapter, referred to as the Siebel EAI eWay throughout this guide.

**What's in This Chapter**



## 1.1 About the Siebel EAI eWay

The Siebel EAI eWay enables the Sun Java Composite Application Platform Suite to exchange messages with the Siebel EAI interface via a Web server using open standards such as HTTP and XML. There are two distinct processes involved in using the Siebel EAI eWay:

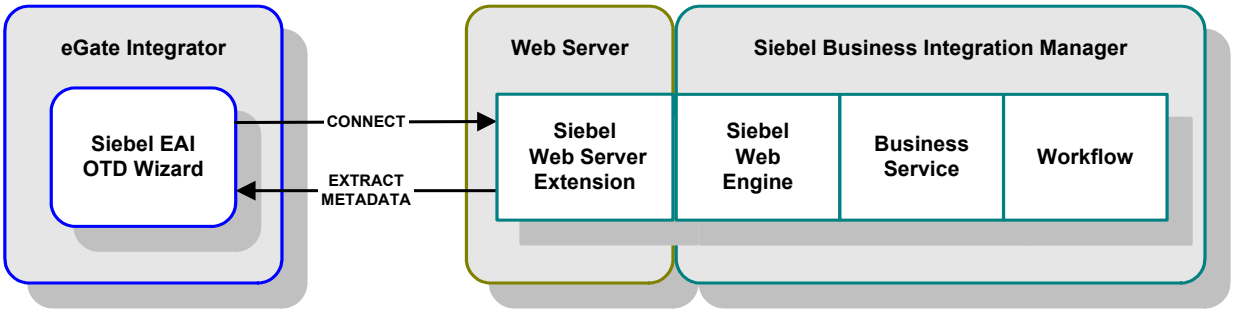
- A design-time process, in which you obtain information about the Siebel Interface Object; and
- A run-time process, in which you use the Project to exchange data with Siebel EAI.

### 1.1.1 Design-Time Process

The design-time process, which is an integral part of Project development, is primarily concerned with extracting metadata from the Siebel application. This metadata is then used to format the messages propagated by the eWay.

This process uses the Siebel EAI OTD Wizard, which prompts you for information to find and connect to the desired Siebel instance. The Wizard then connects to Siebel (see Figure 1) and extracts the business services that are exposed through the Siebel Web Engine. These services are presented to you for selection of the appropriate service and operation.

**Figure 1** Metadata Extraction




When the service and operation have been selected, an OTD representing the selections is generated and saved in the repository.

## 1.1.2 Run-Time Process

During run-time, the Siebel EAI eWay's components relay the contents of web requests to Java Collaborations or Business Processes for further processing and subsequent hand-off to an outbound Siebel EAI eWay.

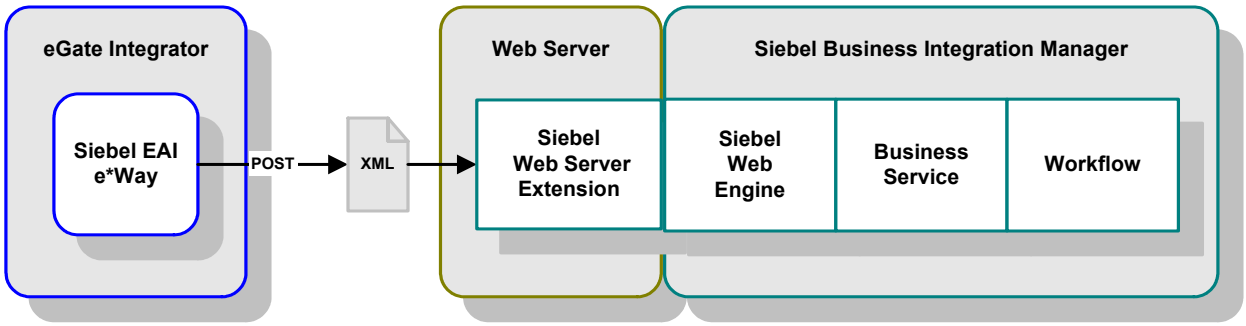
In routine operation, the Siebel EAI eWay uses HTTP to post a Siebel XML-formatted message to Siebel. It also specifies one of the following actions to be performed on the XML message:

- Delete
- Upsert (Insert/Update)
- Query

The result is that a corresponding Workflow is executed to process the message. A Siebel Workflow is a customized business application for managing and enforcing business processes.

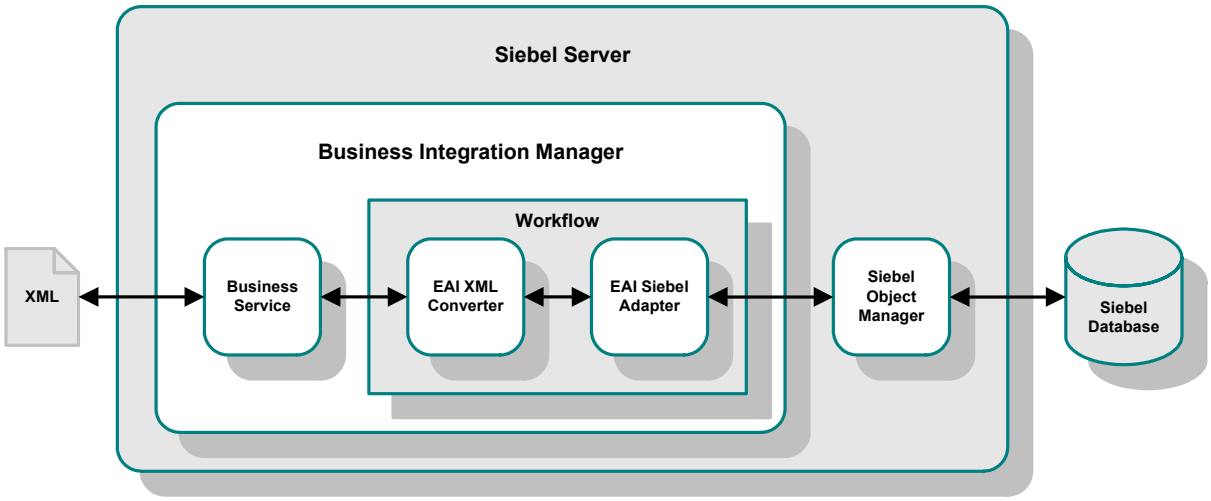
The Siebel EAI eWay POSTs the message to the Web server. The Siebel Web Server Extension invokes the specified Business Service which, in turn, starts an internal Workflow. Figure 2 illustrates the process.

**Figure 2** Java CAPS to Siebel Data Flow



The Workflow invokes the Siebel EAI XML Converter, which converts the information from XML into the Siebel internal format and presents it to the Siebel EAI Adapter. The information is then sent to the Siebel Server via the Siebel Object Manager (see Figure 3).

**Figure 3** Siebel Internal Processing - POST




If any data is to be returned, the EAI Siebel Adapter can pass the result to the EAI XML Converter and send the data back to the eWay as a Siebel XML message.

## 1.1.3 Workflow Templates

A set of Workflow Templates is included with the Siebel EAI eWay. These Workflow Templates invoke the necessary Workflow Processes to map the data directly to or from the Siebel database. See **"Setting Up Siebel Workflows" on page 24**.

## 1.1.4 Session vs. Sessionless Mode

You can run the Siebel EAI eWay in either session or sessionless mode. When running in the default Sessionless mode, every message posted to Siebel is enveloped with the login method, negating the need for an explicit login. By contrast, when Siebel runs in Session mode, the collaboration must include both a login method at beginning and a logout method at the end. Session mode allows you to post multiple messages to Siebel within a loop between a single login and logoff statement. Session mode is only supported using the Java Collaboration Definition (JCD). You cannot use Session mode when using business processes in eInsight.

## 1.1.5 Using the Siebel Message Header

Siebel EAI eWay supports both Siebel integration objects and Application Service Interfaces (ASIs). A Siebel message header is required for most integration objects or ASIs. In a JCD, you can include the Siebel Message Header by invoking the appropriate methods provided in the Siebel EAI OTD. When creating business processes in

eInsight, the Siebel Message Header is automatically included when the appropriate web service operation (Query, Update, Insert, Delete) is selected. Also, be sure to set the integrationObjectName.

## 1.2 What's New in This Release

The Sun SeeBeyond eWay Adapter for Siebel EAI includes the following changes and new features:

**New for Version 5.1.2**

- This is a maintenance release. No new features.

**New for Version 5.1.1**

- This is a maintenance release. No new features.

**New for Version 5.1.0**

- Version Control: An enhanced version control system allows you to effectively manage changes to the eWay components.
- Multiple Drag-and-Drop Component Mapping from the Deployment Editor: The Deployment Editor now allows you to select multiple components from the Editor's component pane, and drop them into your Environment component.
- Support to Obtain Configuration from LDAP at Runtime: eWay configuration properties now support LDAP key values.
- Relaunchable OTD Support: An OTD can be rebuilt and saved (under the same name), and used in the same Java Collaboration or BPEL. This allows you to rebuild the OTD with changed metadata without having to completely recreate the JCD or BPEL.
- Connectivity Map Generator: Generates and links your Project's Connectivity Map components using a Collaboration or Business Process.

Many of these features are documented further in the *Sun SeeBeyond eGate™ Integrator User's Guide* or the *Sun SeeBeyond eGate™ Integrator System Administration Guide*.

## 1.3 About This Document

This document includes the following chapters:

- **Chapter 1 "Introducing the Siebel EAI eWay"**: Provides an overview description of the product as well as high-level information about this document.
- **Chapter 2 "Installing the Siebel EAI eWay"**: Describes the system requirements and provides instructions for installing the Siebel EAI eWay.
- **Chapter 3 "Setting Up the Web Server"**: Describes the procedures for configuring a Web Server to connect the Siebel application with the Siebel EAI eWay.

- **Chapter 4 "Setting Up Siebel Workflows"**: Describes the procedures for setting up the SeeBeyond Workflow Templates.
- **Chapter 5 "Configuring the Siebel EAI eWay"**: Provides instructions for configuring the eWay to communicate with your Siebel system.
- **Chapter 6 "Using the Siebel EAI OTD Wizard"**: Provides instructions for creating Object Type Definitions to be used with the Siebel EAI eWay.
- **Chapter 7 "Implementing the Siebel EAI eWay Sample Projects"**: Provides instructions for installing and running the sample Projects.

### Siebel EAI eWay Javadoc

A Siebel EAI eWay Javadoc is also provided that documents the Java methods available with the Siebel EAI eWay. The Javadoc is uploaded with the eWay's documentation file (**SiebelEAIeWayDocs.sar**) and downloaded from the **Documentation** tab of the Sun Java Composite Application Platform Suite Installer. To access the full Javadoc, extract the Javadoc to an easily accessible folder, and double-click the **index.html** file.

## 1.3.1 Scope

This user's guide provides a description of the Siebel EAI eWay Adapter. It includes directions for installing the eWay, configuring the eWay properties, and implementing the eWay's sample Projects. This document is also intended as a reference guide, listing available properties, functions, and considerations. For a reference of available Siebel EAI eWay Java methods, see the associated Javadoc.

## 1.3.2 Intended Audience

This guide is intended for experienced computer users who have the responsibility of helping to set up and maintain a fully functioning Java Composite Application Platform Suite system. This person must also understand any operating systems on which the Java Composite Application Platform Suite will be installed (Windows and UNIX), and must be thoroughly familiar with Windows-style GUI operations.

This guide also assumes that the reader has a strong working knowledge of the Siebel system, and is experienced in using and configuring the Siebel system and Siebel Workflows.

## 1.3.3 Text Conventions

The following conventions are observed throughout this document.

**Table 1** Text Conventions

| Text Convention | Used For | Examples |
|---|---|---|
| **Bold** | Names of buttons, files, icons, parameters, variables, methods, menus, and objects | ▪ Click **OK**.<br>▪ On the **File** menu, click **Exit**.<br>▪ Select the **eGate.sar** file. |

**Table 1** Text Conventions (Continued)

| Text Convention | Used For | Examples |
|---|---|---|
| `Monospaced` | Command line arguments, code samples; variables are shown in ***bold italic*** | `java -jar ***filename***.jar` |
| **Blue bold** | Hypertext links within document | See **Text Conventions** on page 11 |
| Blue underlined | Hypertext links for Web addresses (URLs) or email addresses | http://www.sun.com |

## 1.4 Related Documents

The following Sun documents provide additional information about the Sun Java Composite Application Platform Suite product:

- *Sun SeeBeyond eGate™ Integrator User's Guide*
- *Sun Java Composite Application Platform Suite Installation Guide*

## 1.5 Sun Microsystems, Inc. Web Site

The Sun Microsystems web site is your best source for up-to-the-minute product news and technical support information. The site's URL is:

http://www.sun.com

## 1.6 Documentation Feedback

We appreciate your feedback. Please send any comments or suggestions regarding this document to:

CAPS_docsfeedback@sun.com

Chapter 2

# Installing the Siebel EAI eWay

**What's in This Chapter**



## 2.1 Siebel EAI eWay System Requirements

The Siebel EAI eWay Readme contains the latest information on:

- Supported Operating Systems
- System Requirements
- External System Requirements

The Siebel EAI eWay Readme is uploaded with the eWay's documentation file (**SiebelEAIeWayDocs.sar**) and can be accessed from the **Documentation** tab of the Sun Java Composite Application Platform Suite Installer. Refer to the Siebel EAI eWay Readme for the latest requirements before installing the Siebel EAI eWay.

## 2.2 Installing the Siebel EAI eWay

The Sun Java Composite Application Platform Suite Installer, a web-based application, is used to select and upload eWays and add-on files during the installation process. The following section describes how to install the components required for this eWay.

**Note:** *When the Repository is running on a UNIX operating system, the eWays are loaded from the Sun Java Composite Application Platform Suite Installer running on a Windows platform connected to the Repository server using Internet Explorer.*

### 2.2.1 Installing the Siebel EAI eWay on an eGate supported system

Follow the directions for installing the Sun Java Composite Application Platform Suite (CAPS).

**After you have installed eGate or eInsight, do the following:**

1. From the Enterprise Manager's **ADMINISTRATION** tab, click on **license** in the Sun Java Composite Application Platform Suite Products Installed table. The Sun Java Composite Application Platform Suite Web Installer appears.
2. From **Select Sun Java Composite Application Platform Suite Products to Install**, select the products for your **Sun Java Composite Application Platform Suite** and include the following:
   - **FileeWay** (the File eWay is used by most sample Projects)
   - **SiebelEAIeWay**

   To upload the Sun SeeBeyond eWay™ Siebel EAI Adapter User's Guide, Help file, Javadoc, Readme, and sample Projects, select the following:
   - **SiebelEAIeWayDocs**
3. Once you have selected all of your products, click **Next** in the top-right or bottom-right corner of the **Select Sun Java Composite Application Platform Suite Products to Install** box.
4. From the **Selecting Files to Install** box, locate and select your first product's SAR file. Once you have selected the SAR file, click **Next**. Follow this procedure for each of your products. The Installing Files window appears after the last SAR file has been selected.
5. From the **Installing Files** window, review the product list. If it is correct, Click **Install Products**. The Enterprise Manager starts the installation.
6. When your product's installation is completed, click on the prompt, "**When installation completes, click here to continue**."
7. Continue installing the eGate Integrator as instructed in the *Sun Java Composite Application Platform Suite Installation Guide*.

## Adding the eWay to an Existing Sun Java Composite Application Platform Suite Installation

It is possible to add the eWay to an existing Sun Java Composite Application Platform Suite installation.

**Steps required to add an eWay to an Existing CAPS installation include:**

1. Complete steps 1 through 6 on **"Installing the Siebel EAI eWay on an eGate supported system" on page 13**.
2. Open the Enterprise Designer and select **Update Center** from the Tools menu. The Update Center Wizard appears.
3. For Step 1 of the wizard, simply click **Next**.

4 For Step 2 of the wizard, click the **Add All** button to move all installable files to the **Include in Install** field, then click **Next**.

5 For Step 3 of the wizard, wait for the modules to download, then click **Next**.

6 The wizard's Step 4 window displays the installed modules. Review the installed modules and click **Finish.**

7 When prompted, restart the IDE (Integrated Development Environment) to complete the installation.

### After Installation

Once you install the eWay, it must then be incorporated into a Project before it can perform its intended functions. See the *eGate Integrator User's Guide* for more information on incorporating the eWay into an eGate Project.

## 2.2.2 Extracting the Sample Projects and Javadocs

The Siebel EAI eWay includes sample Projects and Javadocs. The sample Projects are designed to provide you with a basic understanding of how certain operations are performed using the eWay, while Javadocs provide a list of classes and methods exposed in the eWay.

**Steps to extract the Javadoc include:**

1 Click the **Documentation** tab of the Sun Java Composite Application Platform Suite Installer, then click the Add-ons tab.

2 Click the Siebel EAI eWay Adapter link. Documentation for the Siebel EAI eWay appears in the right pane.

3 Click the icon next to **Javadoc** and extract the ZIP file.

4 Open the index.html file to view the Javadoc.

**Steps to extract the Sample Projects include:**

1 Click the **Documentation** tab of the Sun Java Composite Application Platform Suite Installer, then click the Add-ons tab.

2 Click the Siebel EAI eWay Adapter link. Documentation for the Siebel EAI eWay appears in the right pane.

3 Click the icon next to **Sample Projects** and extract the ZIP file. Note that the **SiebelEAI_eWay_Sample.zip** file contains an additional ZIP files for each sample Project (**SiebelEAI_bpUpsert.zip** and **SiebelEAI_jcdUpsert.zip**).

Refer to **"Importing a Sample Project" on page 62** for instructions on importing the sample Project into your repository via the Enterprise Designer.

# 2.3 ICAN 5.0 Project Migration Procedures

This section describes how to transfer your current ICAN 5.0.x Projects to the Sun Java Composite Application Platform Suite 5.1.2. To migrate your ICAN 5.0.x Projects to the Sun Java Composite Application Platform Suite 5.1.2, do the following:

### Export the Project

1. Before you export your Projects, save your current ICAN 5.0.x Projects to your Repository.
2. From the Project Explorer, right-click your Project and select **Export** from the shortcut menu. The Export Manager appears.
3. Select the Project that you want to export in the left pane of the Export Manager and move it to the Selected Projects field by clicking the **Add to Select Items** (arrow) button, or click **All** to include all of your Projects.
4. In the same manner, select the Environment that you want to export in the left pane of the Export Manager and move it to the Selected Environments field by clicking the **Add to Select Items** (arrow) button, or click **All** to include all of your Environments.
5. Browse to select a destination for your Project ZIP file and enter a name for your Project in the **ZIP file** field.
6. Click **Export** to create the Project ZIP file in the selected destination.

### Install Java CAPS 5.1.2

1. Install the **Java CAPS 5.1.2**, including all eWays, libraries, and other components used by your ICAN 5.0 Projects.
2. Start the Java CAPS 5.1.2 Enterprise Designer.

### Import the Project

1. From the Java CAPS 5.1.2 Enterprise Designer's Project Explorer tree, right-click the Repository and select **Import Project** from the shortcut menu. The Import Manager appears.
2. Browse to and select your exported Project file.
3. Click **Import**. A warning message, **"Missing APIs from Target Repository**," may appear at this time. This occurs because various product APIs were installed on the ICAN 5.0 Repository when the Project was created, that are not installed on the Java CAPS 5.1.2 Repository. These APIs may or may not apply to your Projects. You can ignore this message if you have already installed all of the components that correspond to your Projects. Click **Continue** to resume the Project import.
4. Close the Import Manager after the Project is successfully imported.

### Deploy the Project

1. A new Deployment Profile must be created for each of your imported Projects. When a Project is exported, the Project's components are automatically *"checked in"* to Version Control to write-protected each component. These protected components appear in the Explorer tree with a red padlock in the bottom-left corner of each icon.

Before you can deploy the imported Project, the Project's components must first be *"checked out"* of Version Control from both the Project Explorer and the Environment Explorer. To *"check out"* all of the Project's components, do the following:

A From the Project Explorer, right-click the Project and select **Version Control > Check Out** from the shortcut menu. The Version Control - Check Out dialog box appears.

B Select **Recurse Project** to specify all components, and click **OK**.

C Select the Environment Explorer tab, and from the Environment Explorer, right-click the Project's Environment and select **Version Control > Check Out** from the shortcut menu.

D Select **Recurse Environment** to specify all components, and click **OK**.

2 If your imported Project includes File eWays, these must be reconfigured in your Environment prior to deploying the Project.

To reconfigure your File eWays, do the following:

A From the Environment Explorer tree, right-click the File External System, and select **Properties** from the shortcut menu. The Properties Editor appears.

B Set the inbound and outbound directory values, and click **OK**. The File External System can now accommodate both inbound and outbound eWays.

3 Deploy your Projects.

**Note:** *Only projects developed on ICAN 5.0.2 and later can be imported and migrated successfully into the Java Composite Application Platform Suite.*

## 2.4 Installing Enterprise Manager eWay Plug-Ins

The **Sun SeeBeyond Enterprise Manager** is a Web-based interface you use to monitor and manage your Java Composite Application Platform Suite applications. The Enterprise Manager requires an eWay specific "plug-in" for each eWay you install. These plug-ins enable the Enterprise Manager to target specific alert codes for each eWay type, as well as start and stop the inbound eWays.

The *Sun Java Composite Application Platform Suite Installation Guide* describes how to install Enterprise Manager. The *Sun SeeBeyond eGate Integrator System Administration Guide* describes how to monitor servers, Services, logs, and alerts using the Enterprise Manager and the command-line client.

The **eWay Enterprise Manager Plug-ins** are available from the **List of Components to Download** under the Sun Java Composite Application Platform Suite Installer's **DOWNLOADS** tab.

There are two ways to add eWay Enterprise Manager plug-ins:

- From the **Sun SeeBeyond Enterprise Manager**
- From the **Sun Java Composite Application Platform Suite Installer**

**To add plug-ins from the Enterprise Manager**

1. From the **Enterprise Manager**'s Explorer toolbar, click **configuration**.
2. Click the **Web Applications Manager** tab, go to the **Auto-Install from Repository** sub-tab, and connect to your Repository.
3. Select the application plug-ins you require, and click **Install**. The application plug-ins are installed and deployed.

**To add plug-ins from the Sun Java Composite Application Platform Suite Installer**

1. From the **Sun Java Composite Application Platform Suite Installer**'s **Download** tab**,** select the Plug-Ins you require and save them to a temporary directory.
2. From the **Enterprise Manager**'s Explorer toolbar, click **configuration**.
3. Click the **Web Applications Manager** tab and go to the **Manage Applications** sub-tab.
4. Browse for and select the WAR file for the application plug-in that you downloaded, and click **Deploy**. The plug-ins is installed and deployed.

## 2.4.1 Viewing Alert Codes

You can view alerts using the Enterprise Manager. An alert is triggered when a specified condition occurs in a Project component. The purpose of the alert is to warn the administrator or user that a condition has occurred.

**To View the eWay Alert Codes**

1. Add the eWay Enterprise Manager plug-in for this eWay.
2. From the **Enterprise Manager**'s Explorer toolbar, click the **Configuration** icon.
3. Click the **Web Applications Manager** tab and go to the **Manage Alert Codes** tab. Your installed eWay alert codes display under the **Results** section. If your eWay alert codes are not displayed under **Results**, do the following:
   A. From the **Install New Alert Codes** section, browse to and select the eWay alert properties file for the application plug-in that you added. The alert properties files are located in the **alertcodes** folder of your Sun Java Composite Application Platform Suite installation directory.
   B. Click **Deploy**. The available alert codes for your application are displayed under **Results**. A listing of the eWay's available alert codes is displayed in Table 2.

**Table 2** Alert Codes for the Siebel EAI eWay

| Alert Code\Description | Description Details | User Actions |
|---|---|---|
| SIEBELEAIEWAY-LOGIN-FAILED000002=Failed to login to url {0} with user name {1}. | Occurs when a login is attempted at an invalid URL or with an incorrect username or password. | ▪ Verify that the URL is valid.<br>▪ Verify that the username and password association is valid.<br>▪ Access the Siebel server log for further information. |

| Alert Code\Description | Description Details | User Actions |
|---|---|---|
| SIEBELEAIEWAY-LOGOFF-FAILED000003=Failed to log off from Siebel server at url {0} with user name {1}. | Occurs when a logoff is attempted. | ▪ Access the Siebel server log for further information. |
| SIEBELEAIEWAY-POSTFORM-FAILED000001=Failed to post to url {0} with SWEExtSource {1}, SWEExtCmd {2}, and SWEExtData {3}. | Occurs when a Siebel operation is not successful. | ▪ Access the Siebel server log for further information. |
| SIEBELEAIEWAY-REQFAIL001=An incoming request could not be processed. Event: [{0}] Detail: [{1}] | Occurs when a Siebel operation is not successful. | ▪ Access the Siebel server log for further information. |
| SIEBELEAIWAY-CONNECT-FAILED000004=Failed to prepare Siebel EAI Client to establish connection to Siebel EAI server. | Occurs during initial Siebel system connection establishment. | ▪ Verify that the configuration parameters for the Siebel client are correct.<br>▪ Verify that the Siebel server is running. |
| SIEBELEAIWAY-POST-FAILED000006=Failed to handle the Siebel EAI POST request to URL {0}. | Occurs when a Siebel post operation is not successful. | ▪ Read the response code in the collaboration and proceed accordingly. |
| SIEBELEAIWAY-URL-FAILED000005=Invalid Siebel URL specified {0}. | Occurs when an invalid URL is entered. | Verify that the URL is correct. |

For information on Managing and Monitoring alert codes and logs, as well as how to view the alert generated by the project component during runtime, see the *Sun SeeBeyond eGate™ Integrator System Administration Guide.*

**Note:** *An alert code is a warning that an error has occurred. It is not a diagnostic. The user actions noted above are just some possible corrective measures you may take. Refer to the log files for more information. For information on Managing and Monitoring alert codes and logs, see the Sun SeeBeyond eGate Integrator System Administration Guide.*

Chapter 3

# Setting Up the Web Server

This chapter describes procedures for configuring a Web Server to connect the Siebel application with the Siebel EAI eWay.

**What's in This Chapter**



## 3.1 Overview

Communication with the Siebel 7.x EAI application involves one of the following Web servers:

- Microsoft Internet Information Server (IIS) for Windows
- iPlanet Web Server for Solaris
- IBM HTTP Server (Limited Release) for AIX

**Note:** *This is a special release of IBM HTTP Server 2.0. See the Siebel Systems Requirements and Supported Platforms documentation for information.*

Certain components from Siebel must be installed into the Web server to complete the communications link. See **"Installing the Siebel Web Server Extension" on page 21**

### 3.1.1 Preliminary Installations

The following software must be in place and operating correctly:

- Siebel 7.x
- Siebel Tools
- Siebel Server
- Microsoft IIS, iPlanet Web Server, or IBM HTTP Server (see above)

# 3.2 Installing the Siebel Web Server Extension

The Siebel Web Server Extension (SWSE) is a shared library that runs inside the Web server to direct user requests to the appropriate Application Object Manager service via the Siebel Web Engine (SWE).

### To Install Siebel Web Server Extension (SWSE)

1. From the Siebel installation media, run **\eappweb\setup.exe**, which invokes the installation wizard.
2. Follow the instructions presented by the wizard. Use the naming conventions for your Siebel EAI Application Server.
3. For **Connection Protocol**, specify the default port for an HTTP server, which is **80**.
4. Do *not* use any encryption or compression methods.
5. For **Anonymous Employee** and **Anonymous Contact** login and password, use **SADMIN**.
6. For **Error Level for Logging**, enter **All Errors and Warnings**. You can change this once correct system operation has been verified.
7. In the **\bin** directory where you have installed the Siebel Web Server Extension, open the **eapps.cfg** file and note the following (typical values are shown):

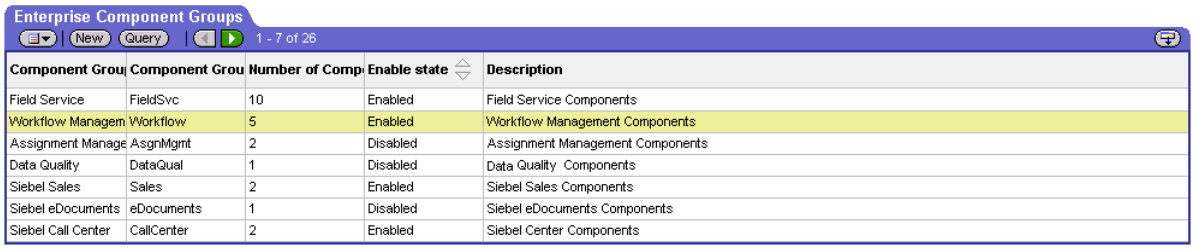
```
[defaults]
AnonUserName = SADMIN
AnonPassword = SADMIN
AnonUserPool = 10
StatsPage = _stats.swe

[/eai]
ConnetString = siebel.TCPIP.none.none://MyGatewayServer:2320/
MyEnterpriseServer/eaiObjMgr/MyAppServer
EnableExtServiceOnly = TRUE
```

8. After modifying these files, stop and then restart the following services:
   - Siebel Server
   - World Wide Web Publishing Service
9. Log in to Siebel 7 and follow the **Screens** menu path:

   **Ctrl+Shift+A > Server Administration > Enterprise Configuration > Enterprise Component Groups**

**Figure 4** Enterprise Component Groups




| Component Grou | Component Grou | Number of Comp | Enable state | Description |
|---|---|---|---|---|
| Field Service | FieldSvc | 10 | Enabled | Field Service Components |
| Workflow Managem | Workflow | 5 | Enabled | Workflow Management Components |
| Assignment Manage | AsgnMgmt | 2 | Disabled | Assignment Management Components |
| Data Quality | DataQual | 1 | Disabled | Data Quality Components |
| Siebel Sales | Sales | 2 | Enabled | Siebel Sales Components |
| Siebel eDocuments | eDocuments | 1 | Disabled | Siebel eDocuments Components |
| Siebel Call Center | CallCenter | 2 | Enabled | Siebel Center Components |

10 Select the following items, and click **Enable**:

- Enterprise Application Integration
- Workflow Management
- Communication Management

11 Open the browser and type:

```
<yourservername>/<module> (for example,10.1.3.135/eai)
```

and then click **Enter**.

Make sure that your Siebel Server ODBC data source is configured. You can verify which one you are using by examining the Siebel Server log directory—it contains a file listing all the parameters.

### To Verify SWSE Operation for Siebel 7

1 Verify that the **Server Request Processor** is running correctly. You may need to synchronize the Server Request Components with the Gateway Server—follow the menu/command path:

**Enterprise Configuration > Batch Components Admin > Synchronize**

2 Verify that the **EAI Object Manager** is running correctly.

A Follow the path:

**Server Admin > Servers > Server Components**

B Under **Assignment Components**, select **EAI Object Manager**.

3 *For Windows installations only*, verify that the **.swe** file is associated with **sweiis.dll** in the web site. Use the following procedure to set the association:

A Run the IIS 4.0 **Management Console** application.

B Right-click on your Web site entry in the tree display, and select **Properties**.

C Select the **Home Directory** tab.

D In the **Application Settings** box, select **Configuration**.

E On the **App Mappings** tab, select **Add**.

F Type **swe**, select **sweiis.dll**, and click **OK** to save the association.

4 Verify that the configuration files are set up properly. If LDAP is not used, comment out all LDAP-related parameters in the configuration files (for example, **eai.cfg** and **siebel.cfg**):

```
;[SecurityAdapters]
;LDAP = LDAP

;[LDAP]
;DllName = sscfldap.dll
;ServerName =
;Port = 389
;BaseDN =
;UsernameAttributeType = uid
;PasswordAttributeType = userPassword
;CredentialsAttributeType = credentials
;RolesAttributeType = roles
```

```
;SslDatabase =
```

**Note:** *The LDAP configuration may differ in your Siebel system depending on your version of Siebel and your installed options. For more information on disabling LDAP in your Siebel system, consult your Siebel user documentation.*

5 After modifying these files, stop and then restart the following services:

- Siebel Server
- World Wide Web Publishing Service

Chapter 4

# Setting Up Siebel Workflows

This chapter describes procedures for setting up the SeeBeyond Workflow Templates.

**What's in This Chapter**



## 4.1 Overview

Included with the Siebel EAI eWay are several SeeBeyond Workflow Templates, which furnish pre-defined workflows within the Siebel application. These workflow templates invoke the following workflow processes to map the data directly to or from the Siebel database.

- **SeeBeyond HTTP Delete** (see **Figure 6 on page 26**)
- **SeeBeyond HTTP Query** (see **Figure 7 on page 26**)
- **SeeBeyond HTTP Update** (see **Figure 8 on page 27**)

  Inserts or Updates according to the provided input values.
- **SeeBeyond HTTP Execute** (see **Figure 9 on page 27**)

  The preferred Workflow for receiving Siebel XML messages from Java CAPS; combines **Delete**, **Query** and **Update** functionality into a single Workflow.
- **SeeBeyond** HTTP Send (see **Figure 10 on page 28**)
- **SeeBeyond HTTP Send Receive** (see **Figure 11 on page 28**)
- **SeeBeyond HTTP Post** (see **Figure 12 on page 29**)

  The preferred Workflow for sending Siebel XML messages to Java CAPS; combines **Send** and **Send/Receive** functionality into a single Workflow.

The names entered above are used to set up the Business Service for the sample program supplied with the eWay. You should use them as templates to create new processes corresponding to the Workflows you create for your own system.

Examples of XML messages used with different Workflow templates are given in **"Siebel XML Messages" on page 39**.

### Restarting the Gateway and Siebel Server

Most of the configuration steps described in this chapter require you to stop the Gateway and the Siebel Server, copy the newly compiled **SWF** file(s) to the Siebel Server, and restart the Gateway and Siebel Server.

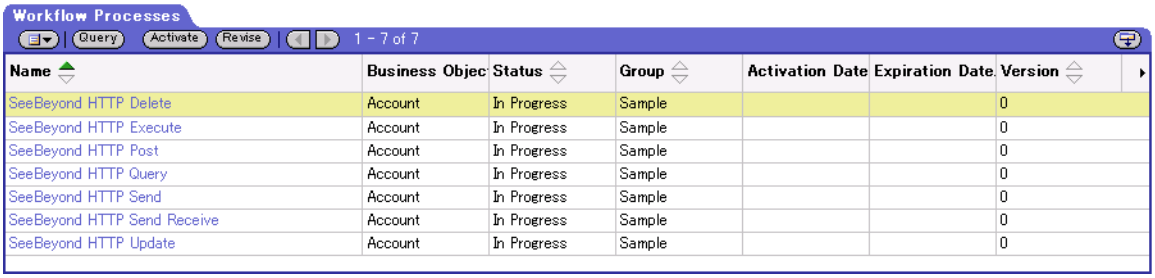
Any time you modify and compile a **SWF** file, you *must* restart the Gateway and the Siebel Server. Restarting the Gateway and Siebel Server is intentionally omitted from many of the procedures in this chapter, because it is assumed that you will only want to restart the server one time—after you have completed all of the steps in this chapter.

## 4.2 Viewing the Workflow Templates

Following the Screens menu path shown below displays the installed Workflow templates, as shown in Figure 5.

Siebel Workflow Administration > Workflow Processes > All Processes

**Figure 5** SeeBeyond Workflow Processes




Clicking the process name to invoke a Workflow Process Designer display for that process, such as shown in the figures below.

**Figure 6** DELETE Workflow Template

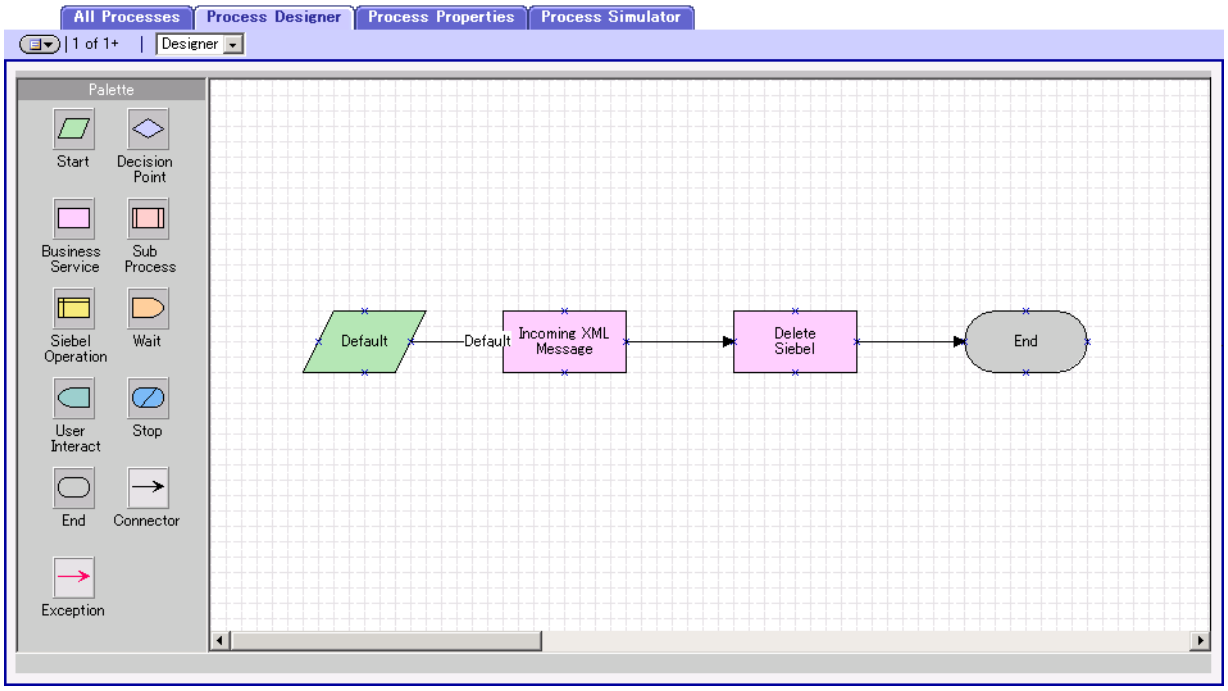



**Figure 7** QUERY Workflow Template

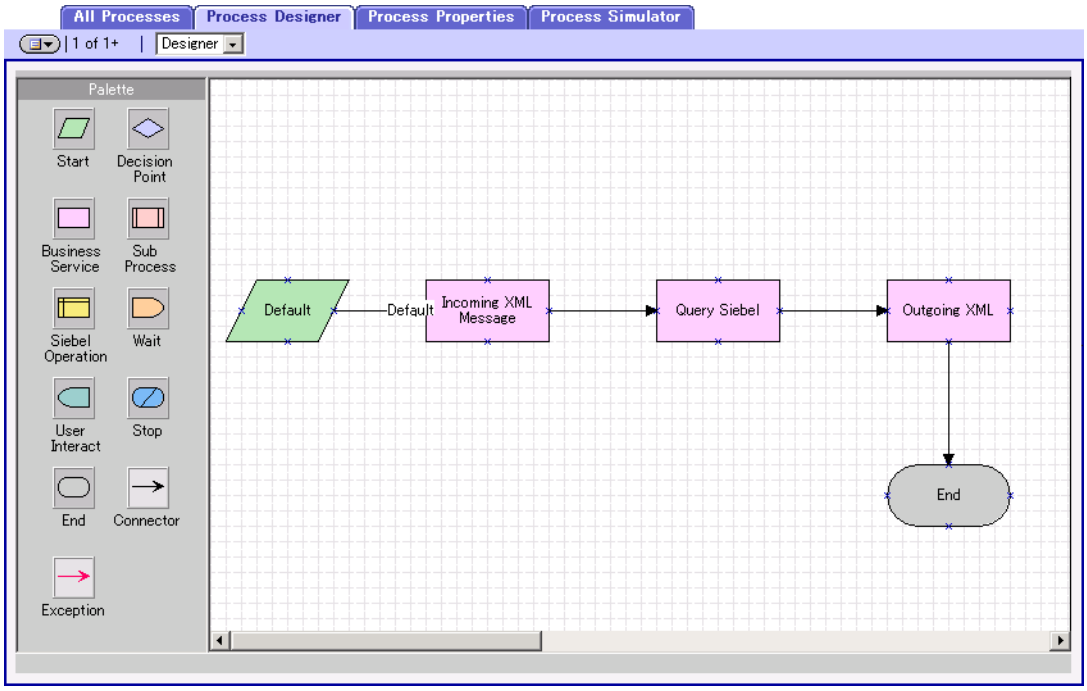


**Figure 8** INSERT/UPDATE Workflow Template

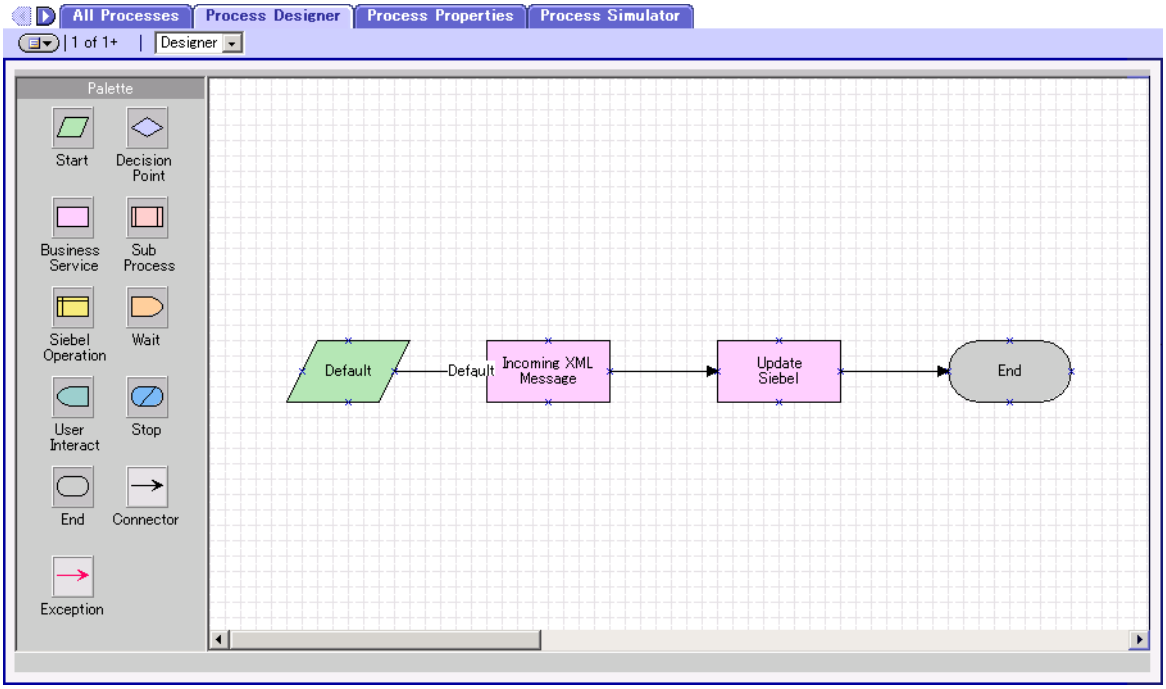



**Figure 9** EXECUTE Workflow Template

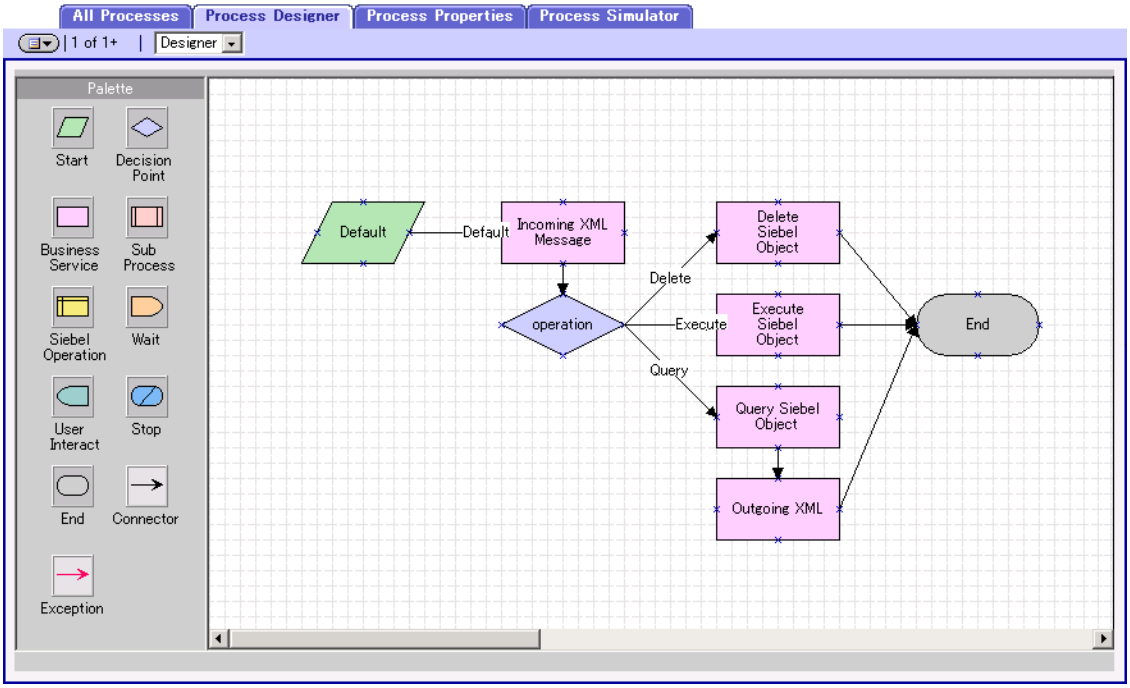


**Figure 10** SEND Workflow Template

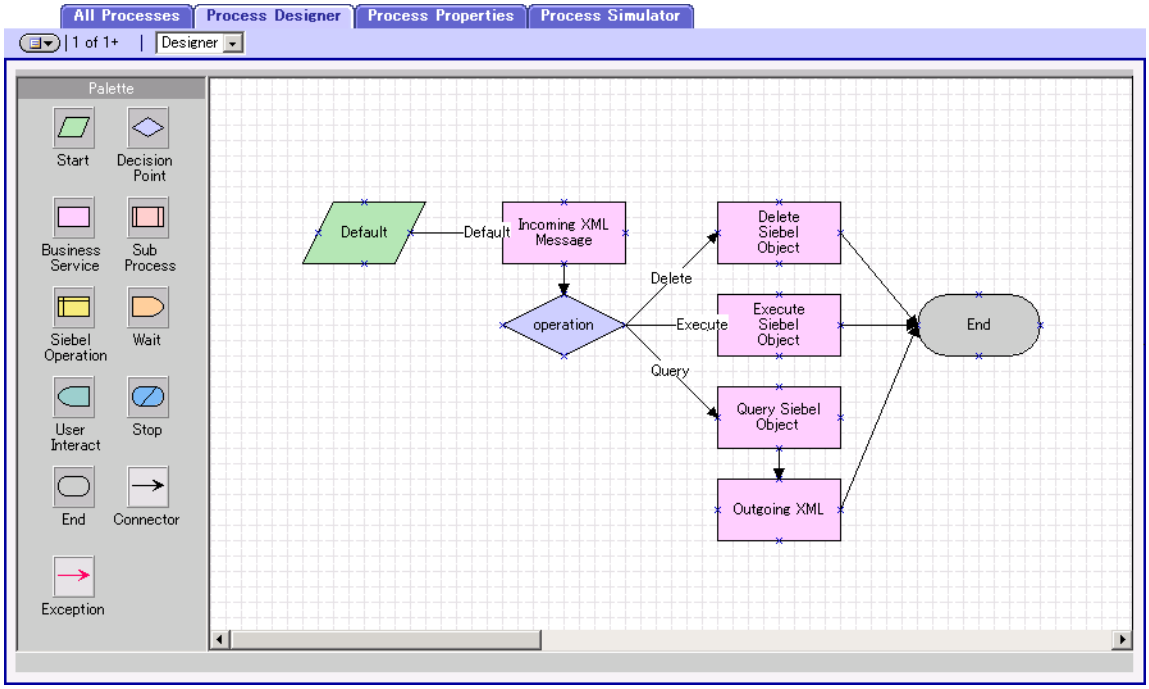



**Figure 11** SEND/RECEIVE Workflow Template

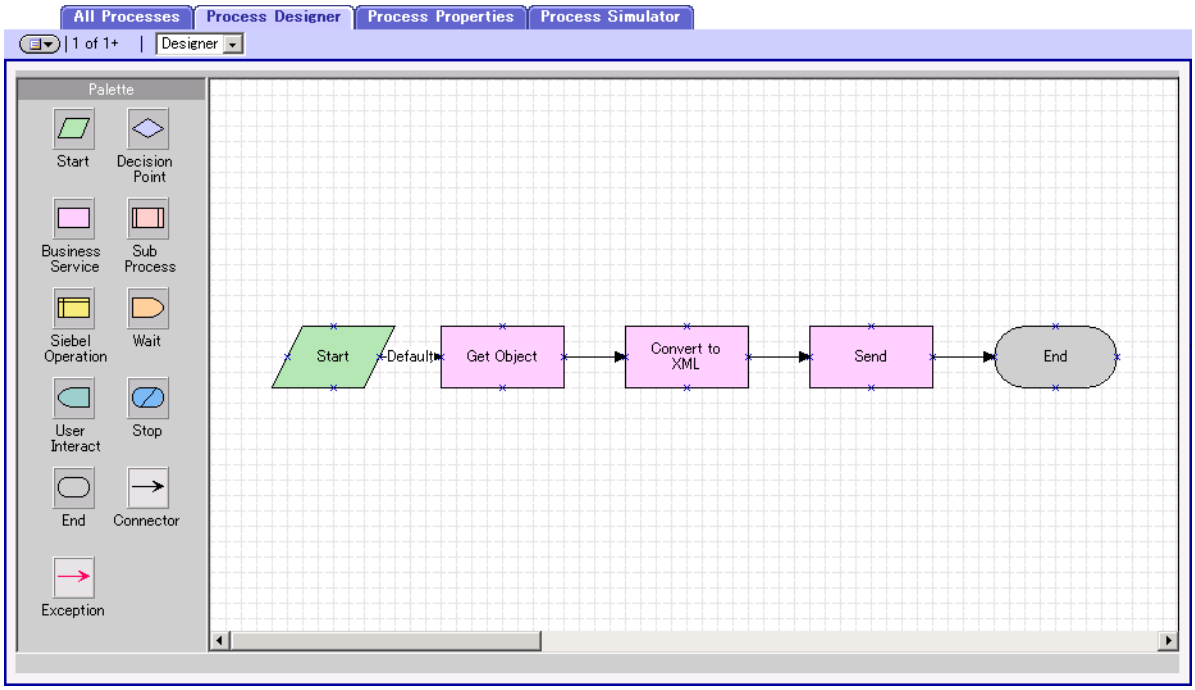
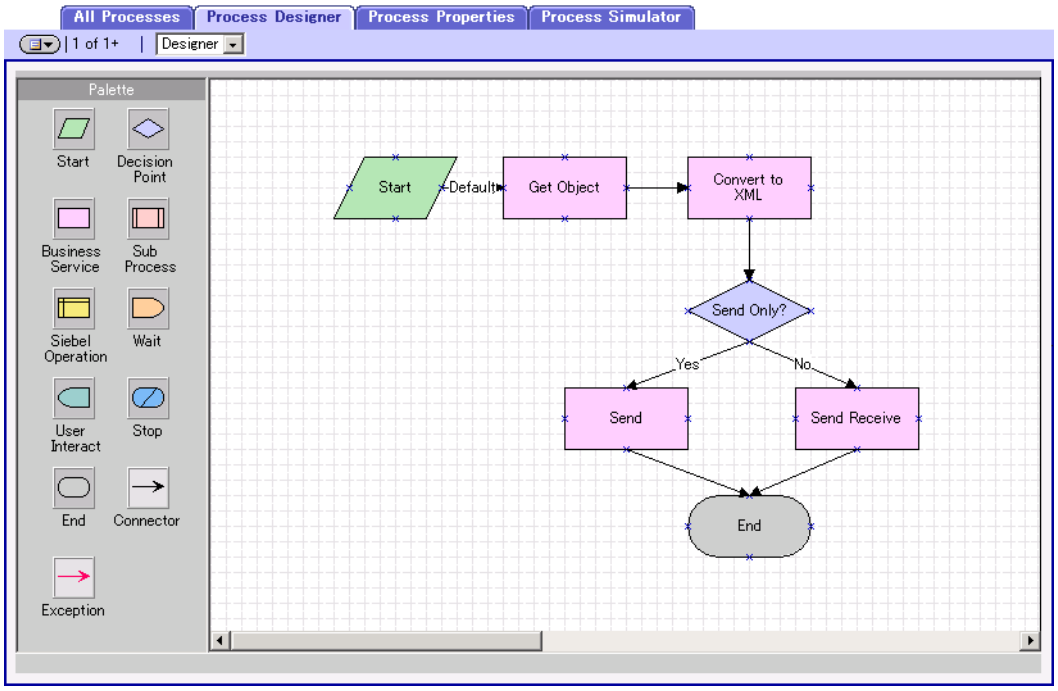


**Figure 12** POST Workflow Template




## 4.3 Importing SeeBeyond Workflow Templates

**To Import the SeeBeyond Workflow Templates with Siebel 7.5**

1. Start **Siebel EAI Client** and select **Siebel Sales**.
2. Follow the menu path:

   Ctrl+Shift+A > Siebel Workflow Administration > Workflow Processes
3. Click **Import** and browse to the directory that contains **SeeBeyondHTTPWorkflow.xml**.

   This file is installed with the Enterprise Designer. If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then the file is located in

   ```
   <C:\JavaCAPS>\edesigner\usrdir\modules\ext\siebeleaieway
   ```
4. Click **Open** to begin importing the Workflow template.
5. Check or set up the following configuration files:
   - In the file **SWEApp\eapps.cfg**, verify that the following section is specified correctly:

     ```
     [/eai_enu]
     ConnectString = siebel.TCPIP.none.none://<Your Gateway
     Server>:2320/<Your_Enterprise_Server>/EAIObjMgr_enu/<Your_App
     Server>
     EnableExtServiceOnly = TRUE
     WebPublicRootDir=c:\<Your_Siebel_Dir>\SWEApp\public\enu
     ```

```
WebUpdatePassword=<Your_Password>
```

- For the **SiebelEAI_jcdUpsert** sample, add the following sections in the file **siebsrvr\bin\ENU\eai.cfg**:

```
[HTTP Services]
...
SEEBEYOND_HTTP_DELETE = SB_HTTP_DELETE
SEEBEYOND_HTTP_QUERY = SB_HTTP_QUERY
SEEBEYOND_HTTP_UPDATE = SB_HTTP_UPDATE
SEEBEYOND_HTTP_EXECUTE = SB_HTTP_EXECUTE

[SB_HTTP_DELETE]
Mode = Document
Service = SeeBeyond HTTP Delete
Method = RunProcess

[SB_HTTP_QUERY]
Mode = Document
Service = SeeBeyond HTTP Query
Method = RunProcess

[SB_HTTP_UPDATE]
Mode = Document
Service = SeeBeyond HTTP Update
Method = RunProcess
[SB_HTTP_EXECUTE]
Mode = Document
Service = SeeBeyond HTTP Execute
Method = RunProcess
```

6 The imported Workflow must be in *active* mode before it can be used; this is accomplished by clicking the **active** button.

### To Import the SeeBeyond Workflow Templates with Siebel 7.7

1 Open the **Siebel Tools** utility.

2 In the Object Explorer, click the Workflow Process.

3 Right-click the item in the Object list, and then click **Import Workflow Process**.

4 Browse to the directory that contains the workflow files shown in Table 3.

These files are installed with the Enterprise Designer. If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then the files are located in

```
c:\JavaCAPS\edesigner\usrdir\modules\ext\siebeleaieway
```

**Table 3** Siebel 7.7 Workflow Files

| File Name |
|---|
| SeeBeyond HTTP Delete.xml |
| SeeBeyond HTTP Execute.xml |
| SeeBeyond HTTP Post.xml |
| SeeBeyond HTTP Query.xml |
| SeeBeyond HTTP Send Receive.xml |
| SeeBeyond HTTP Send.xml |
| SeeBeyond HTTP Update.xml |

**Table 3** Siebel 7.7 Workflow Files (Continued)

| File Name |
|---|
| Sync Account ASI Inbound.xml |

5 Click **Open** to begin importing the Workflow template. Repeat this step for each of the workflow files you plan to install.

6 Check or set up the following configuration files:

- In the file **SWEApp\eapps.cfg**, verify that the following section is specified correctly:

```
[/eai_enu]
ConnectString = siebel.TCPIP.none.none://<Your Gateway
Server>:2320/<Your_Enterprise_Server>/EAIObjMgr_enu/<Your_App
Server>
EnableExtServiceOnly = TRUE
WebPublicRootDir=c:\<Your_Siebel_Dir>\SWEApp\public\enu
WebUpdatePassword=<Your_Password>
```

- For the **SiebelEAI_jcdUpsert** sample, add the following sections in the file **siebsrvr\bin\ENU\eai.cfg**:

```
SEEBEYOND_HTTP_UPDATE = SEEBEYOND_HTTP_UPDATE
SEEBEYOND_HTTP_DELETE = SEEBEYOND_HTTP_DELETE
SEEBEYOND_HTTP_QUERY = SEEBEYOND_HTTP_QUERY
SEEBEYOND_HTTP_EXECUTE = SEEBEYOND_HTTP_EXECUTE
```

7 The imported Workflow must be in *active* mode before it can be used; this is accomplished by clicking the **active** button.

## 4.4 Modifying SeeBeyond Workflow Templates

**Note:** *The SeeBeyond Workflow templates provided with the eWay use* **Account** *as the Business Object—you must modify them for use with a different Business Object.*

### To Modify a SeeBeyond Workflow Template

1 Log in to **Siebel Client 7.0**, designating the appropriate Siebel server.

2 Follow the menu path:

Ctrl+Shift+A > Siebel Workflow Administration > Workflow Processes

3 Highlight the SeeBeyond Workflow Process template you want to modify.

4 Right-click and select **Copy Record**.

5 Rename the copied Process.

6 Specify the Business Object to which you want to apply the template, and any other fields that may be necessary (for example, Description).

7 After modifying a Workflow template you must create the Business Service to execute it, using the supplied Workflow processes as templates. This procedure is described in the following section.

8 A new Services section should be added to your **siebsrvr\bin\ENU\eai.cfg** file, as shown in the preceding section.

For example, if you have a Business Service named Employee Execute, you should add the following lines to the **eai.cfg** file:

```
[HTTP Services]
...
EMPLOYEE_EXECUTE = EE

[EE]
Mode = Document
Service = Employee Execute
Method = RunProcess
```

## 4.5 Setting Up SeeBeyond Workflow Processes

The Workflow processes invoked by the SeeBeyond Workflow Templates must be set up in Siebel Business Services.

**Note:** *The names entered in the last step above are used to set up the Business Service for the sample program supplied with the eWay. You should use them as templates to create new processes corresponding to the Workflows you create for your own system.*

### To set up the Business Service to execute the Workflow

1 Make sure the following services are running:

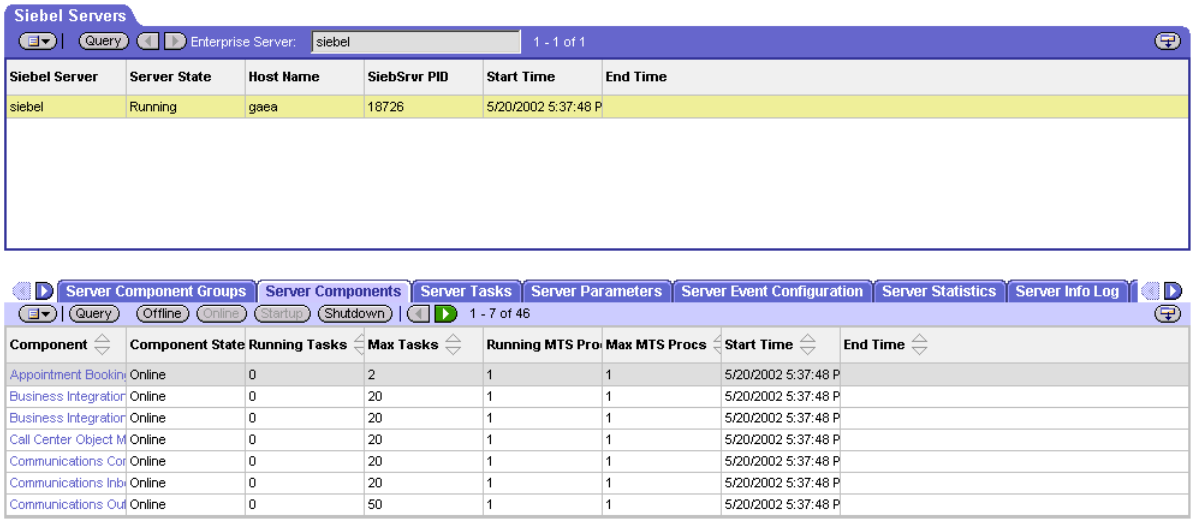
- Siebel Gateway Server
- Siebel Server
- World Wide Web Publishing Service

2 Log in to Siebel Sales 7.0.

3 Follow the **Screens** menu path:

Server Administration > Servers

**Figure 13** Server Component Groups

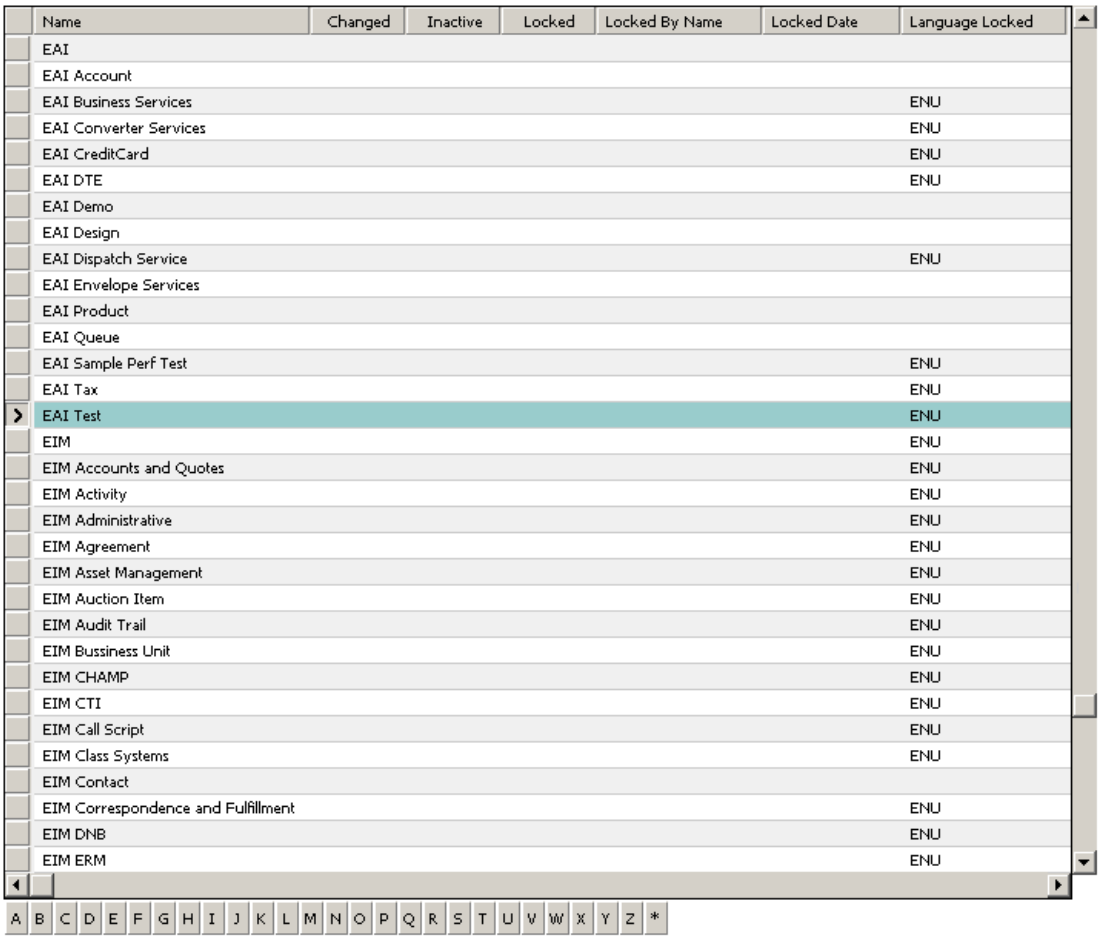



4 Make sure that **Workflow Management** is **Online** and **Enabled**.

5 Log in to Siebel Tools 7.0 and designate the server as the database by entering **sadmin**, **sadmin**, **server**.

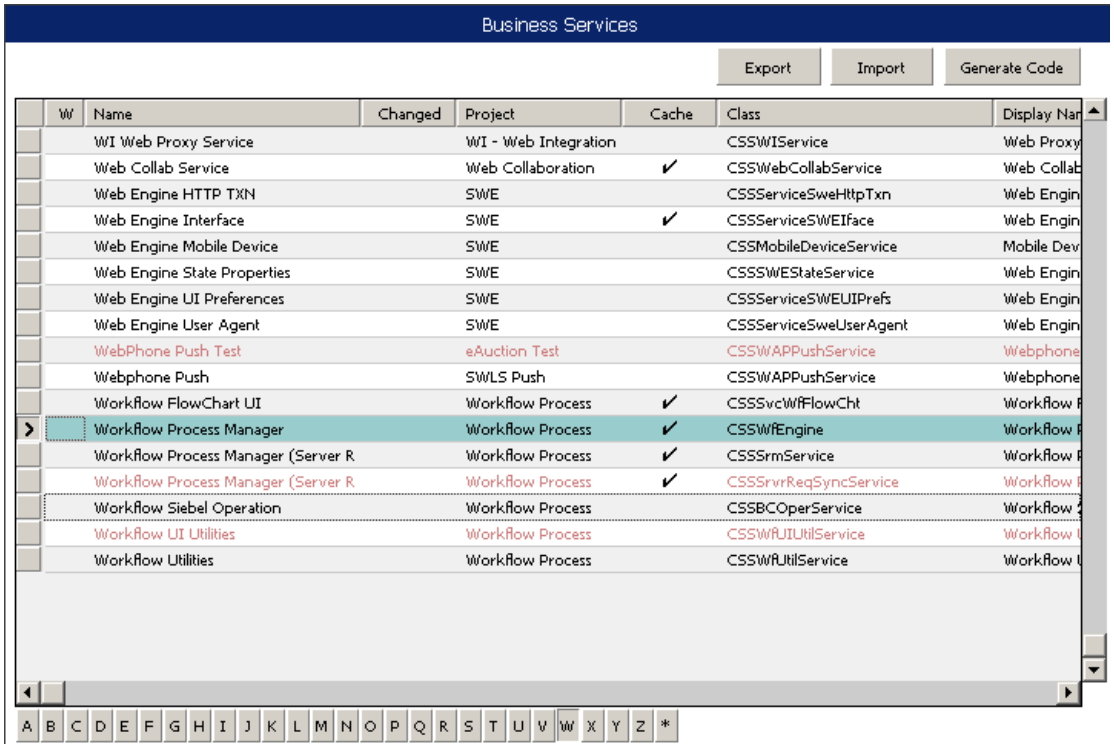
6 In Object Explorer, go to **Siebel Objects** > **Project** and lock the Project (see Figure 14).

**Figure 14** Lock Project




7 In Object Explorer, go to **Business Service**, make a copy of **Workflow Process Manager** (menu path **Edit** > **Copy Record**).

**Figure 15** Business Services View - Workflow Process Manager

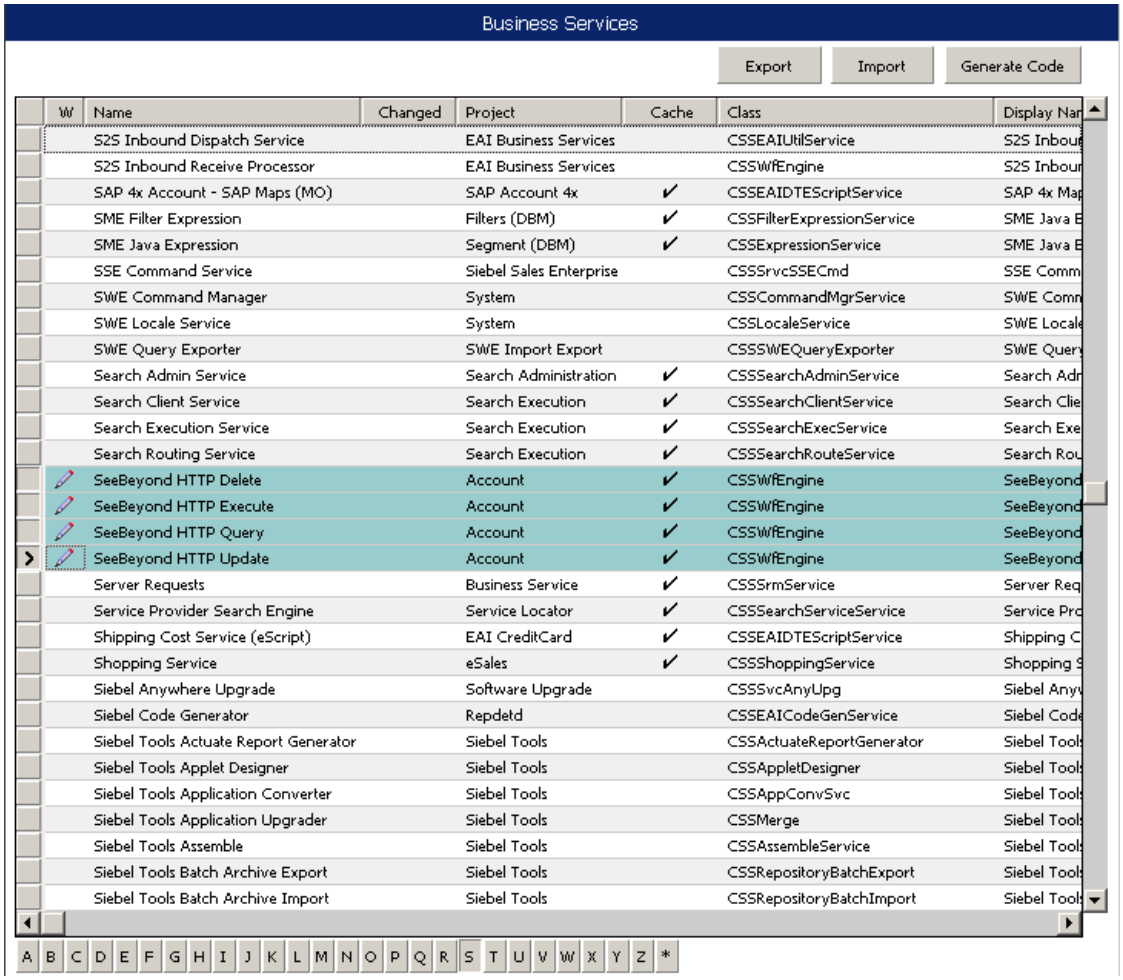



8 Type the Process Name into the **Name** and **Display Names** fields, as shown in Figure 16 (this name is specified in the **eai.cfg** file).

For the **SiebelEAI_jcdUpsert** project, perform this step for the following:

- SeeBeyond HTTP Delete
- SeeBeyond HTTP Execute
- SeeBeyond HTTP Query
- SeeBeyond HTTP Update

**Figure 16** Business Services View - Renamed Fields




9 In Object Explorer, go to **Business Service** > **Business Service Method**:

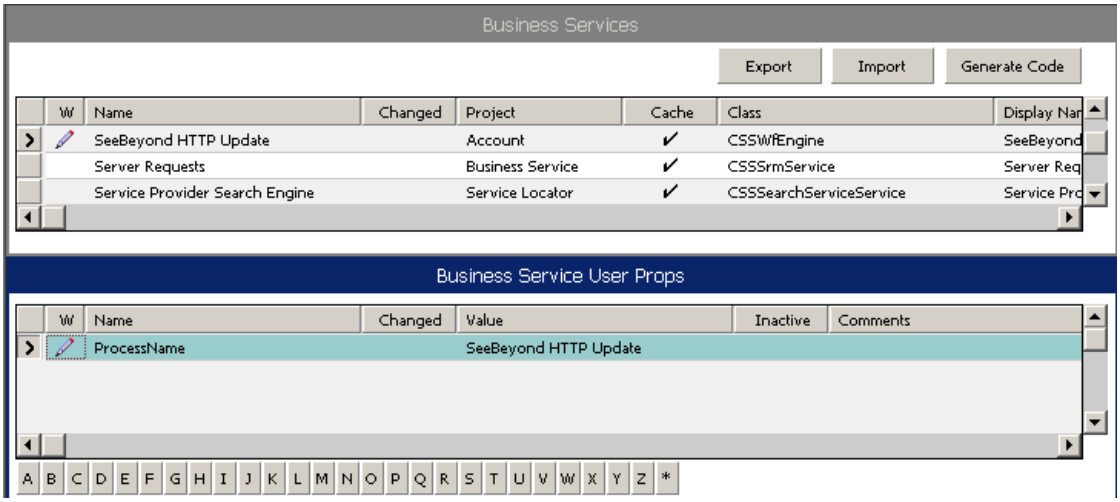
A Type **ProcessName** into the **Name** field.

B Type the actual Process Name into the **Value** field (see Figure 17).

For **SiebelEAI_jcdUpsert** project, perform this step for:

- SeeBeyond HTTP Delete
- SeeBeyond HTTP Execute
- SeeBeyond HTTP Query
- SeeBeyond HTTP Update

**Figure 17** Business Services User Properties



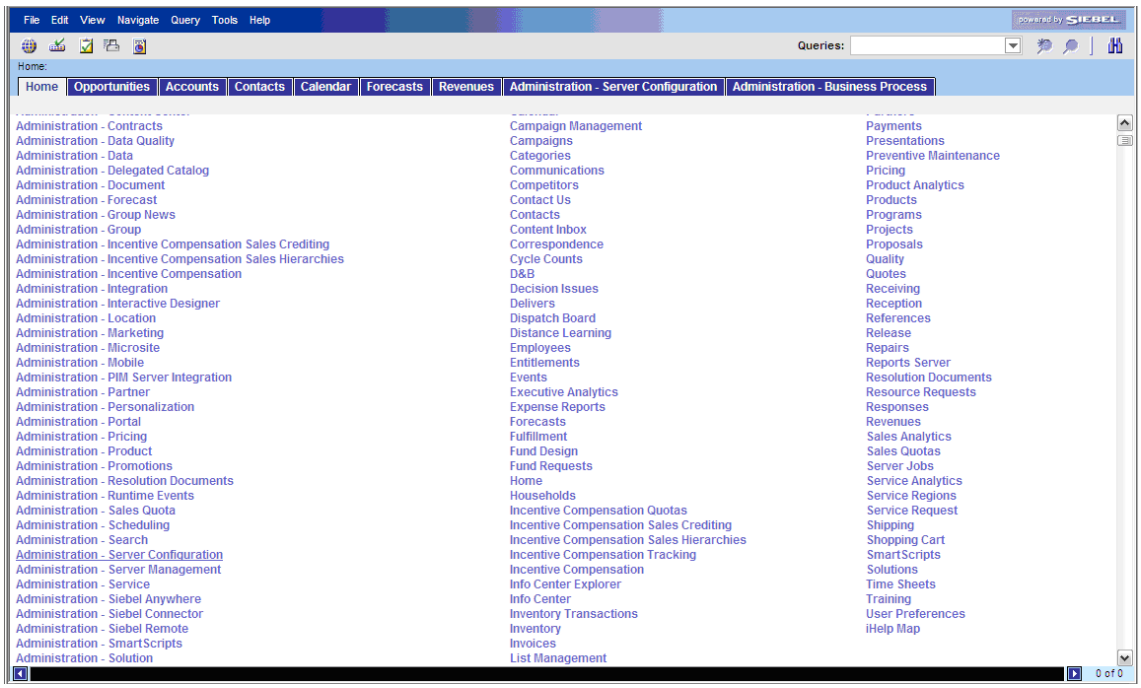

The Workflow Processes you create in the Business Services are similar to those shown in Figure 16.

## 4.6 Creating a Subysystem for a Deployed Workflow

**To Create a Subsystem for a Deployed Workflow:**

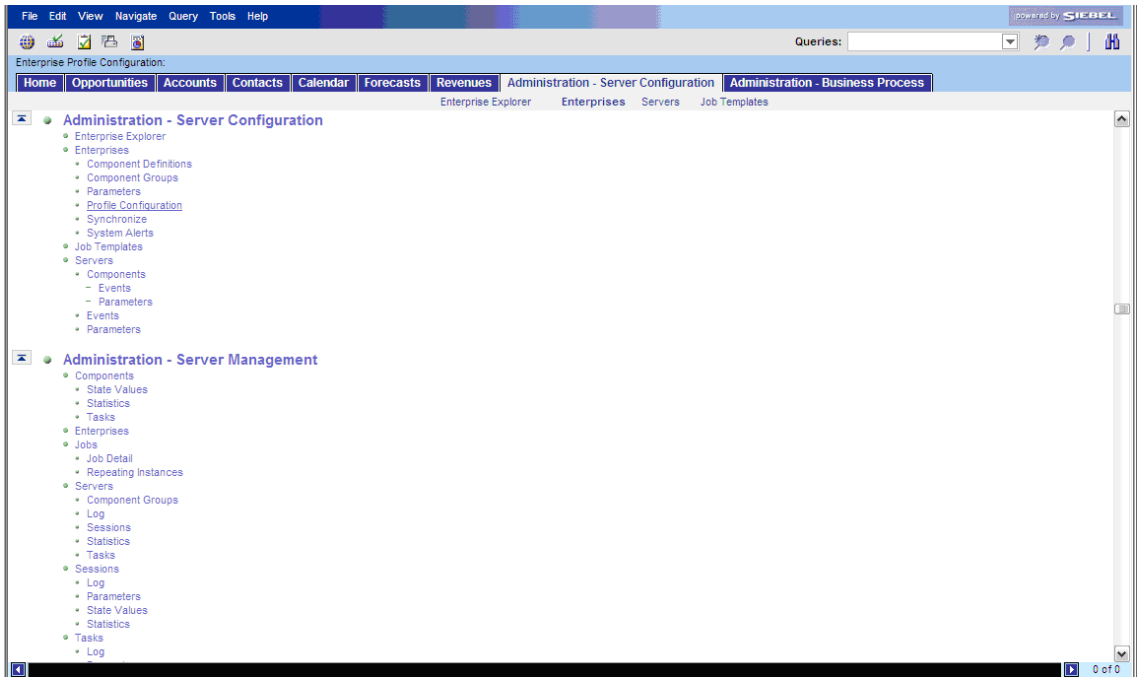
1. Log in to **Siebel Client 7.7**, designating the appropriate Siebel server.
2. Click the Site Map in the Siebel Client home page.

**Figure 18** Administration - Server Configuration




3 Click the **Administration - Server Configuration** link. The **Administration - Server Configuration** page appears.

**Figure 19** Administration - Profile Configuration




4 Click the **Profile Configuration** link.

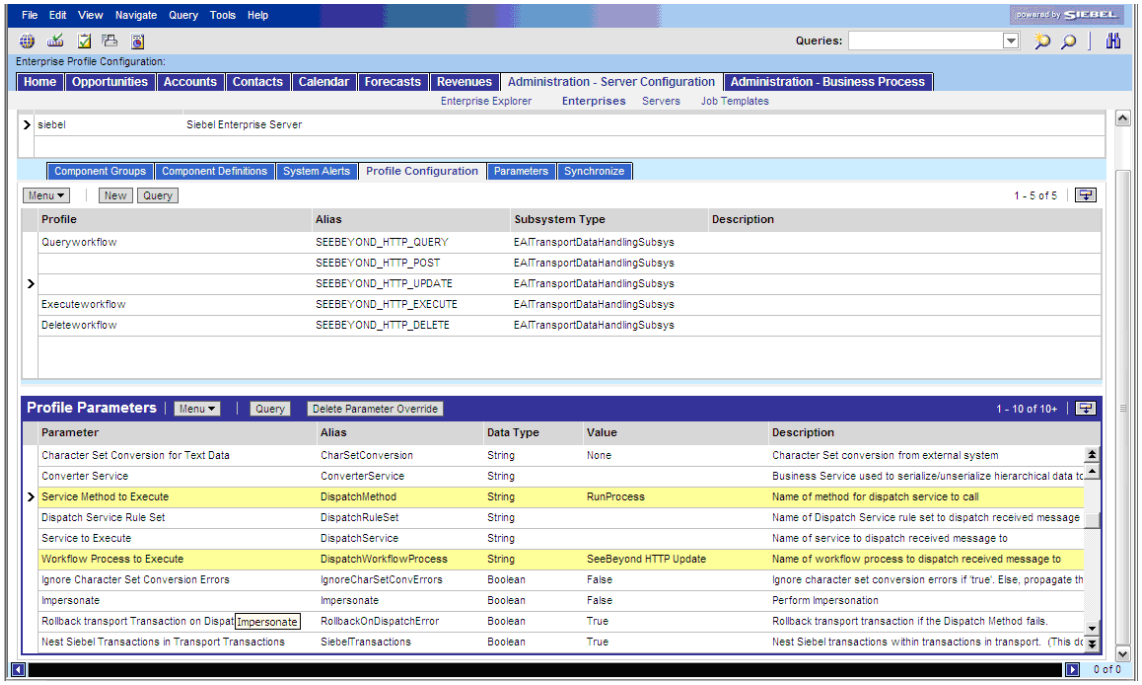
5 In the **Profile Configuration** page, create the Named Subsystem (**SEEBEYOND_HTTP_UPDATE**).

6 Set the **Workflow Process to Execute** parameter for **SEEBEYOND_HTTP_UPDATE** to **SeeBeyond HTTP Update**.

7 Set the **Service Method to Execute** parameter for **SEEBEYOND_HTTP_UPDATE** to **RunProcess**.

For this example, **SEEBEYOND_HTTP_UPDATE** is the name of the Named Subsystem. **SeeBeyond HTTP Update** is the Business Service that invokes the **SeeBeyond HTTP Update** Workflow. See XX.

**Figure 20** SEEBEYOND_HTTP_UPDATE Named Subsystem




# 4.7 Siebel XML Messages

## 4.7.1 Format

A Siebel XML Message used by Siebel EAI eWay has the following format:

Header/Prefix

Integration Object (in XML format)

Footer/Suffix

where:

Header =

```
<SiebelMessage MessageId="" MessageType="Integration Object"
IntObjectName="(Name of Integration Object)" operation=(action)>
```

Footer =

```
</SiebelMessage>
```

and (**action**) can be any of the following values:

- delete
- query
- upsert

**Note:** **operation=(action)** *is used only with the* **EXECUTE** *workflow template.*

## 4.7.2 Examples

### Example 1

The following Siebel XML message specifies that the Integration Object that we are dealing with is **Sample Account**. If we send this message to Siebel EAI using the **INSERT/UPDATE** workflow template, either a new record is generated or an existing record is updated.

```
<SiebelMessage  MessageId=""  MessageType="Integration Object"
IntObjectName="Sample Account">
<ListofSampleAccount>
<Account>
<Name>A. K. Parker Distribution</Name>
<Location>HQ-Distribution</Location>
<Organization>North American Organization</Organization>
<Division></Division>
<CurrencyCode>USD</CurrencyCode>
<Description>This is THE key account in the AK Parker Family</
Description>
<HomePage>www.parker.com</HomePage>
<LineofBusiness>Manufacturing</LineofBusiness>
</Account>
</ListofSampleAccount>
</SiebelMessage>
```

### Example 2

The following Siebel XML message specifies that the Integration Object that we are dealing with is **Sample Account**. If we send this message to Siebel EAI using the **QUERY** workflow template, it returns the object that matches the Name **A. K***

```
<SiebelMessage  MessageId=""  MessageType="Integration Object"
IntObjectName="Sample Account">
<ListofSampleAccount>
<Account>
<Name>A. K*</Name>
</Account>
</ListofSampleAccount>
</SiebelMessage>
```

### Example 3

The following Siebel XML message provides an example of how to use the **operation** attribute with the **Execute** workflow. Here we send the message to Siebel EAI using the **EXECUTE** workflow template to perform a **query** operation. The result is the same as in Example 2.

```
<SiebelMessage  MessageId=""  MessageType="Integration Object"
IntObjectName="Sample Account" operation=query>
<ListofSampleAccount>
<Account>
<Name>A. K*</Name>
</Account>
</ListofSampleAccount>
</SiebelMessage>
```

Chapter 5

# Configuring the Siebel EAI eWay

This chapter describes how to set the properties of the Siebel EAI eWay.

**What's in This Chapter**



## 5.1 Creating and Configuring a Siebel EAI eWay

All eWays contain a unique set of default configuration parameters. After the eWays are established and a Siebel EAI External System is created in the Project's Environment, the eWay parameters are modified for your specific system. The Siebel EAI eWay configuration parameters are modified from two locations:

- **Connectivity Map**: These parameters most commonly apply to a specific component eWay, and may vary from other eWays (of the same type) in the Project.
- **Environment Explorer** : These parameters are commonly global, applying to all eWays (of the same type) in the Project. The saved properties are shared by all eWays in the Siebel EAI External System window.

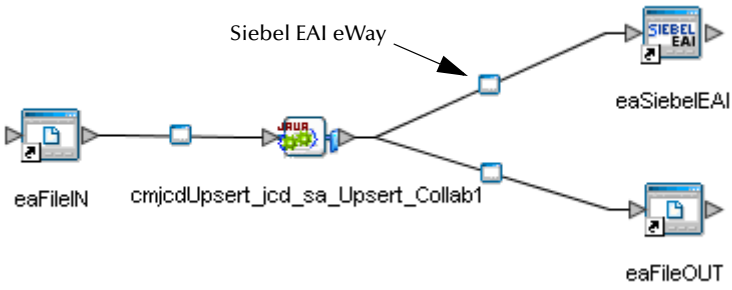
## 5.2 Configuring the eWay Connectivity Map Properties

When you connect an External Application to a Collaboration, Enterprise Designer automatically assigns the appropriate eWay to the link. Each eWay is supplied with a template containing default configuration properties that are accessible on the Connectivity Map.

**To configure the eWay properties:**

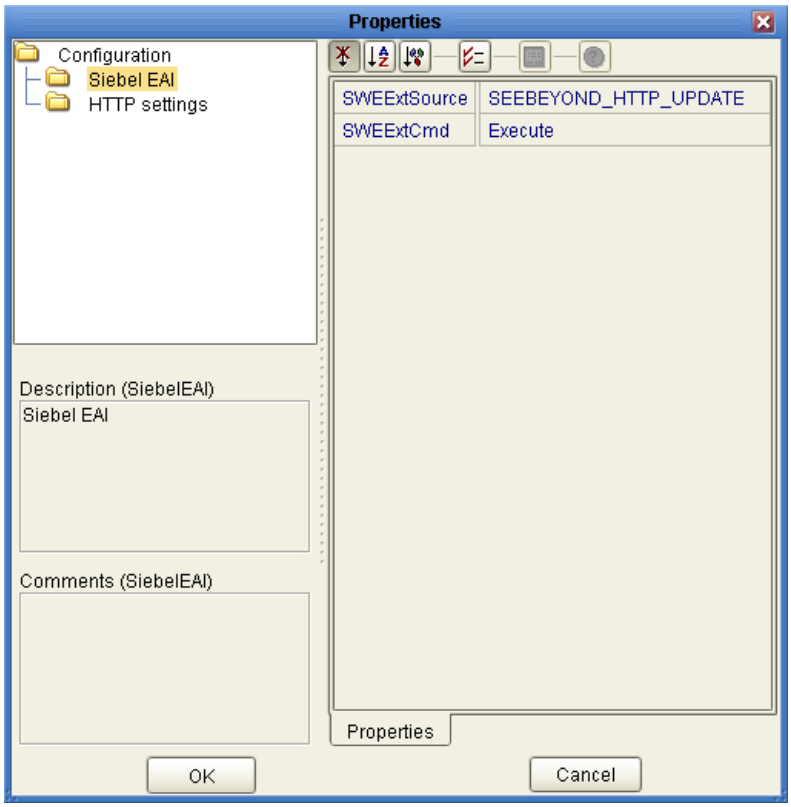
1. On the Enterprise Designer's Connectivity Map, double-click the Siebel EAI eWay icon. The eWay Connections window appears.

**Figure 21** Connectivity Map With Components




The eWay Properties window appears, displaying the default properties for the eWay.

**Figure 22** Outbound eWay Properties




## 5.3 eWay Connectivity Map Properties

The eWay Connectivity Map consists of the following properties categories.

- **Siebel EAI Settings** on page 44
- **HTTP Settings** on page 44

## 5.3.1 Configuring the Connectivity Map eWay Properties

The eWay property settings define how the properties are used to interact with the Siebel application.

### Siebel EAI Settings

Details for Siebel EAI settings are listed in Table 4.

**Table 4** Outbound eWay—Siebel EAI Settings

| Name | Description | Required Value |
|---|---|---|
| SWEExtSource | Specifies the service that the Siebel Web Engine calls. | A valid string. The default is **SEEBEYOND_HTTP_UPDATE**.<br><br>Other values may include:<br>▪ **SEEBEYOND_HTTP_DELETE**<br>▪ **SEEBEYOND_HTTP_EXECUTE**<br>▪ **SEEBEYOND_HTTP_QUERY**<br>▪ **SEEBEYOND_HTTP_UPDATE** |
| SWEExtCmd | Specifies the external command. | A valid entry. The default is **Execute**. |

### HTTP Settings

Details for HTTP settings are listed in Table 5.

**Table 5** Outbound eWay—HTTP Settings

| Name | Description | Required Value |
|---|---|---|
| Allow cookies | Specifies whether or not cookies sent from servers is stored and sent on subsequent requests. If cookies are not allowed, then sessions are not supported. | **True** or **False**. The default is **True**. |
| Content type | Specifies the default Content-Type header value to include when sending a request to the server. | |
| Accept type | Specifies the parameters for the **Accept Type** request header. | Any valid string. For example:<br>▪ text/html<br>▪ text/plain<br>▪ text/xml |
| Encoding | Specifies the default encoding used when reading or writing textual data. | **ASCII**. |

# 5.4 Configuring the eWay Environment Properties

The eWay Environment Configuration properties contain parameters that define how the eWay connects to and interacts with other Java CAPS components within the Environment. When you create a new Siebel EAI External System, you may configure the type of External System required.

Available External System properties include:

- Siebel EAI
- HTTP Settings
- Proxy Configuration
- Security

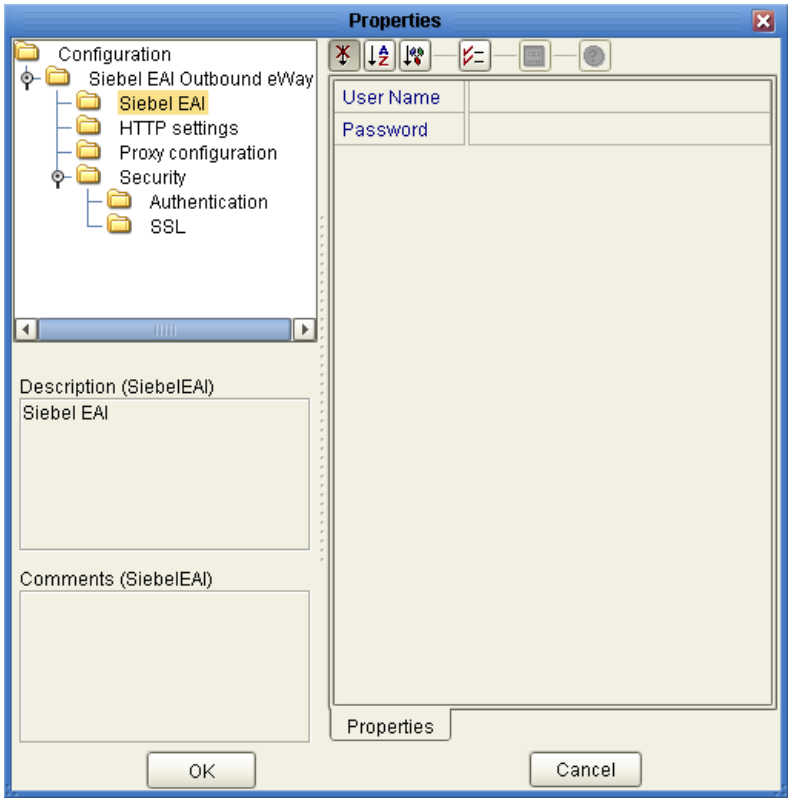
**To Configure the Environment Properties:**

1. In Enterprise Explorer, click the Environment Explorer tab.
2. Expand the Environment created for the Siebel EAI Project and locate the Siebel EAI External System.

**Note:** *For more information on creating an Environment, see the "Sun SeeBeyond eGate Integrator Tutorial".*

3. Right-click the External System created for the Siebel EAI Project and select Properties from the list box. The Environment Configuration Properties window appears.

**Figure 23** Siebel EAI eWay Environment Configuration




4 Click on any folder to display the default configuration properties for that section.

5 Click on any property field to make it editable.

After modifying the configuration properties, click **OK** to save the changes.

## 5.5 eWay Environment Properties

eWay External System properties must be configured from within the Environment. Until you have successfully configured all eWays for your Java CAPS project, your project cannot be properly executed or deployed. The following list identifies the Siebel EAI eWay properties. There are four Environment Configuration categories that the Siebel EAI eWay implements.

### Property Categories Configured in the Logical Host Environment

### 5.5.1 Siebel EAI

Siebel EAI includes the configuration parameters listed in Table 6.

**Table 6** Environment Configuration—Siebel EAI

| Name | Description | Required Value |
|---|---|---|
| User Name | Specifies the user name. | No default value. |
| Password | Specifies the user password. | No default value. |

### 5.5.2 HTTP Settings

HTTP Settings includes the configuration parameters listed in Table 7.

**Table 7** Environment Configuration—HTTP Settings

| Name | Description | Required Value |
|---|---|---|
| URL | Specifies the default URL to be used for establishing an HTTP or HTTPS connection. If HTTPS protocol is specified, SSL must be enabled. | **http://siebel/eai_enu/start.swe**.<br><br>***Important:*** The **URL** property does not support LDAP entries. |

### 5.5.3 Proxy Configuration

Proxy Configuration includes the configuration parameters listed in Table 8.

**Table 8** Environment Configuration—Proxy Configuration

| Name | Description | Required Value |
|---|---|---|
| Proxy Host | The host name of the HTTP proxy. This specifies the HTTPS proxy host to which requests to an HTTP server or reception of data from an HTTP server may be delegated to a proxy. This sets the proxy port for secured HTTP connections. | A valid HTTPS proxy host name. |
| Proxy Port | The port of the HTTPS proxy. | A valid HTTP proxy port. The default is **8080**. |

**Table 8** Environment Configuration—Proxy Configuration (Continued)

| Name | Description | Required Value |
|---|---|---|
| Proxy Username | Specifies the user name required for authentication to access the web site specified by the **URL** property. | A valid user name.<br><br>***Note:*** The user name is required by URLs that require HTTP basic authentication to access the web site.<br><br>***Important:*** Be sure to enter a value for this property before you enter a value for the **Proxy password** properties. |
| Proxy Password | Specifies the password required for authentication to access the web site specified by the **URL** property. | The appropriate password.<br><br>***Important:*** Be sure to enter a value for the **Proxy username** properties before entering this property. |

## 5.5.4 Security

The Environment Configuration Security properties are used to perform HTTP authentication and SSL connections. They include the following configuration sections:

- **Authentication**
- **SSL**

### Authentication

Details for the Authentication settings used for HTTP authentication are detailed in Table 9.

**Table 9** Environment Configuration—Security, Authentication

| Name | Description | Required Value |
|---|---|---|
| HTTP Username | Specifies the user name for authenticating the web site specified by the URL. | A valid user name.<br><br>***Important:*** Enter a value for this property before you enter a value for the **HTTP password** properties. |

**Table 9** Environment Configuration—Security, Authentication (Continued)

| Name | Description | Required Value |
|---|---|---|
| HTTP Password | Specifies the password used for authenticating the web site specified by the URL. | A valid password.<br><br>***Important:*** Be sure to enter a value for the **HTTP username** properties before entering this property. |

## SSL

Details for the SSL settings used for SSL connections are detailed in Table 10.

**Table 10** Environment Configuration—Security, SSL

| Name | Description | Required Value |
|---|---|---|
| Protocol SSL | The SSL protocol to use when establishing an SSL connection with the server. If the protocol is not set by this method, the default protocol type, **TLS** (Sun JSSE), is used. | If you are using the default Sun JSSE provider, choose one of the following settings:<br>♦ **TLSv1**<br>♦ **TLS**<br>♦ **SSLv2**<br>♦ **SSLv3**<br>♦ **SSL**<br>If you are running the Sun SeeBeyond Integration Server on AIX, choose one of the following settings:<br>♦ **SSL-TLS**<br>♦ **TLSv1**<br>♦ **TLS**<br>♦ **SSLv3**<br>♦ **SSLv2**<br>♦ **SSL**<br>For details on these settings, see the appropriate JSSE documentation. |
| JSSE Provider Class | Specifies the fully qualified name of the JSSE provider class. For more information, see the Sun Java Web site at: **http://java.sun.com/** | The name of a valid JSSE provider class. The default is **com.sun.net.ssl.internal.ssl.Provider**<br>If you are running the Sun SeeBeyond Integration Server on AIX, specify **com.ibm.jsse.IBMJSSEProvider**. |

**Table 10** Environment Configuration—Security, SSL (Continued)

| Name | Description | Required Value |
|---|---|---|
| X509 Algorithm Name | Specifies the X509 algorithm name to use for the trust and key manager factories. | The name of a valid X509 algorithm. The default is **SunX509**. If you are running the Sun SeeBeyond Integration Server on AIX, specify **IbmX509**. |
| Verify Hostname | See **Verify hostname** on page 51 for further information. | |
| KeyStore Type | Specifies the default KeyStore type. The keystore type is used for key/certificate management when establishing an SSL connection. If the default KeyStore type is not set by this method, the default KeyStore type, JKS, is used. | |
| KeyStore | Specifies the default KeyStore file. The keystore is used for key/certificate management when establishing SSL connections. | A valid package location. There is no default value. |
| KeyStore Username | The username for accessing the keystore used for key/certificate management when establishing SSL connections.<br><br>***Note:*** If the keystore type is PKCS12 or JKS, the keystore username properties is not used. PKCS12 and JKS keystore types require passwords for access but do not require user names. If you enter a value for this property, it is ignored for PKCS12 and JKS. | |
| KeyStore Password | Specifies the default KeyStore password. The password is used to access the KeyStore used for key/certificate management when establishing SSL connections. | There is no default value. |
| TrustStore Type | The TrustStore type of the TrustStore used for CA certificate management when establishing SSL connections. If the TrustStore type is not set by this method, the default TrustStore type, **JKS**, is used. | A valid **TrustStore** type. |
| TrustStore | Specifies the default TrustStore. The TrustStore is used for CA certificate management when establishing SSL connections. | A valid **TrustStore** name. There is no default value. |

**Table 10** Environment Configuration—Security, SSL (Continued)

| Name | Description | Required Value |
|---|---|---|
| TrustStore Password | Specifies the default TrustStore password. The password is for accessing the TrustStore used for CA certificate management when establishing SSL connections. | A valid **TrustStore** password. There is no default value. |

## 5.5.5 Additional SSL Section Notes

Following are additional notes related to the properties in the SSL section.

### Verify hostname

#### Description

Determines whether the host name verification is done on the server certificate during the SSL handshake.

You can use this property to enforce strict checking of the server host name in the request URL and the host name in the received server certificate.

#### Required Values

**True** or **False**; the default is **False**.

#### Additional information

Under some circumstances, you can get different Java exceptions, depending on whether you set this property to **True** or **False**. This section explains what causes these exceptions.

For example, suppose the host name in the URL is **localhost**, and the host name in the server certificate is **localhost.stc.com**. Then, the following conditions apply:

- If **Verify hostname** is set to **False**:

  Host name checking between the requested URL and the server certificate is turned *off*.

  You can use an incomplete domain host name, for example, **https://localhost:444**, or a complete domain host name, for example, **https://localhost.stc.com:444**, and get a positive response in each case.

  See the next section "Logical Host Java SDK versions" for details.

- If **Verify hostname** is set to **True**:

  Host name checking between the requested URL and the server certificate is turned *on*.

**Note:** *If you use an incomplete domain host name, for example,* **https://localhost:444**, *you can get the exception* **java.io.IOException: HTTPS hostname wrong**.

You must use a complete domain host name, for example, **https://localhost.stc.com:444**.

**Note:** *If the Java Software Developer's Kit (SDK) version used by the Logical Host and the corresponding Logical Host property setting do not match, you can get the exception* **java.lang.ClassCastException**.

Chapter 6

# Using the Siebel EAI OTD Wizard

This chapter describes how to create Object Type Definitions (OTDs) using the Siebel EAI Wizard. OTDs contain the data structure and rules that define an object. They are generated by extracting the business services that have been exposed through the Siebel Web Engine, and the Integration Objects available in the Siebel instance. For the Siebel EAI eWay, this operation is performed by the Siebel EAI Wizard.

**What's in This Chapter**



## 6.1 Before Creating the OTD

### 6.1.1 Installing "seebeyond.sif" for Siebel 7.5.x

If you are using Siebel 7.5.x, before you create an OTD using the OTD Wizard, you must install the **seebeyond.sif** file into your Siebel Tools system. This file is installed in the Enterprise Designer's user directory during the Enterprise Designer installation.

**To install the seebeyond.sif file**

1. Use the Siebel Tools utility to import the **seebeyond.sif** file into your Siebel Server

   This file is installed with the Enterprise Designer. If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then the file is located in

   ```
   <c:\JavaCAPS>\edesigner\usrdir\modules\ext\siebeleaieway
   ```

2. After importing the file, use the Siebel Tools utility to compile your **SRF** file.
3. Stop the Gateway Server and the Siebel Server.
4. Replace the file on the Siebel Server with the one you created in step 2.
5. Restart the Gateway Server and the Siebel Server.

### 6.1.2 Installing "SiebelMessage XSD Generation Process.xml" for Siebel 7.7 and 7.8.x

If you are using Siebel 7.7 or 7.8.x, before you can create an OTD using the OTD Wizard, you must install the **SiebelMessage XSD Generation Process.xml** file.

**To install the "SiebelMessage XSD Generation Process.xml" file**

1. Open the Siebel Tools utility.
2. In the Object Explorer, click the Workflow Process.
3. Right-click the item in the Object list, and then click **Import Workflow Process**.
4. Browse to the directory that contains **SiebelMessage XSD Generation Process.xml**.

   This file is installed with the Enterprise Designer. If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then the file is located in

   ```
   <c:\JavaCAPS>\edesigner\usrdir\modules\ext\siebeleaieway
   ```
5. Click **Open** to begin importing the Workflow template.
6. Select your project.
7. Stop the Gateway Server and the Siebel Server.
8. Use the Siebel Tools utility to compile your **SRF** file.
9. Copy the **SRF** file to the **objects** folder in your Siebel Server.
10. Restart the Gateway Server and the Siebel Server.

## 6.2 Before Running the Enterprise Designer

The Enterprise Designer needs to be configured to use the appropriate JAR files to correspond with the version of your Siebel Server. Before you use the Enterprise Designer to create your Siebel EAI Project, you must make sure that your local Enterprise Designer installation is using the proper JAR files for your Siebel Server.

**To run the Enterprise Designer with Siebel 7.5.x**

1. Using Windows Explorer, navigate to the Siebel eWay directory for your Enterprise Designer.

   If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then navigate to

   ```
   <c:\JavaCAPS>\edesigner\usrdir\modules\ext\siebeleaieway
   ```
2. If you have previously used Siebel 7.7 or 7.8.x with this installation of the Enterprise Designer, then remove the following files from this directory:
   - **Siebel.jar**
   - **SiebelJI_enu.jar**

3 Copy the following files from your Siebel 7.5.x system (SiebelTools/Classes) to this location:

- **SiebelJI.jar**
- **SiebelJI_Common.jar**
- **SiebelJI_enu.jar**

4 Restart the Enterprise Designer

**To run the Enterprise Designer with Siebel 7.7 or 7.8.x**

1 Using Windows Explorer, navigate to the Siebel eWay directory for your Enterprise Designer.

If you installed the Enterprise Designer in the **<c:\JavaCAPS>\edesigner** directory, then navigate to

```
<c:\JavaCAPS>\edesigner\usrdir\modules\ext\siebeleaieway
```

2 If you have previously used Siebel 7.5.x with this installation of the Enterprise Designer, then remove the following files from this directory:

- **SiebelJI.jar**
- **SiebelJI_Common.jar**
- **SiebelJI_enu.jar**

3 Copy the following files from your Siebel 7.7 or 7.8.x system (SiebelTools/Classes) to this location:

- **Siebel.jar**
- **SiebelJI_enu.jar**

4 Restart the Enterprise Designer

## 6.3 Creating the OTD

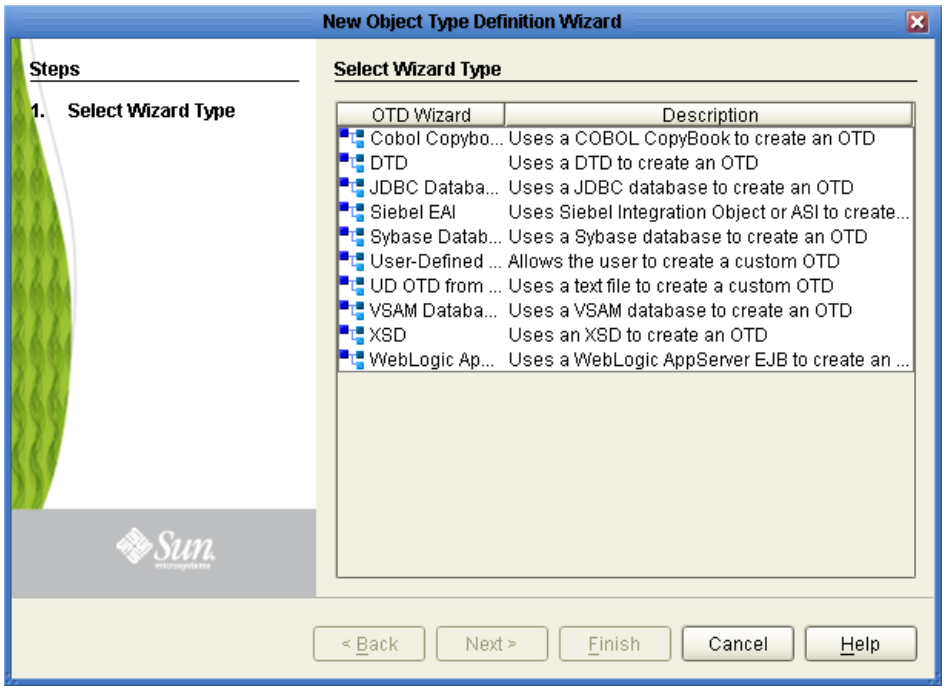
Steps required to create an OTD include:

- **Select Wizard Type**
- **Specify Connection Information**
- **Select Integration Objects**
- **Specify OTD Names**

### Select Wizard Type

On the Project Explorer tree, right click the Project and select **New > Object Type Definition** from the shortcut menu. The **Select Wizard Type** page appears, displaying the available **OTD** wizards. See Figure 24.

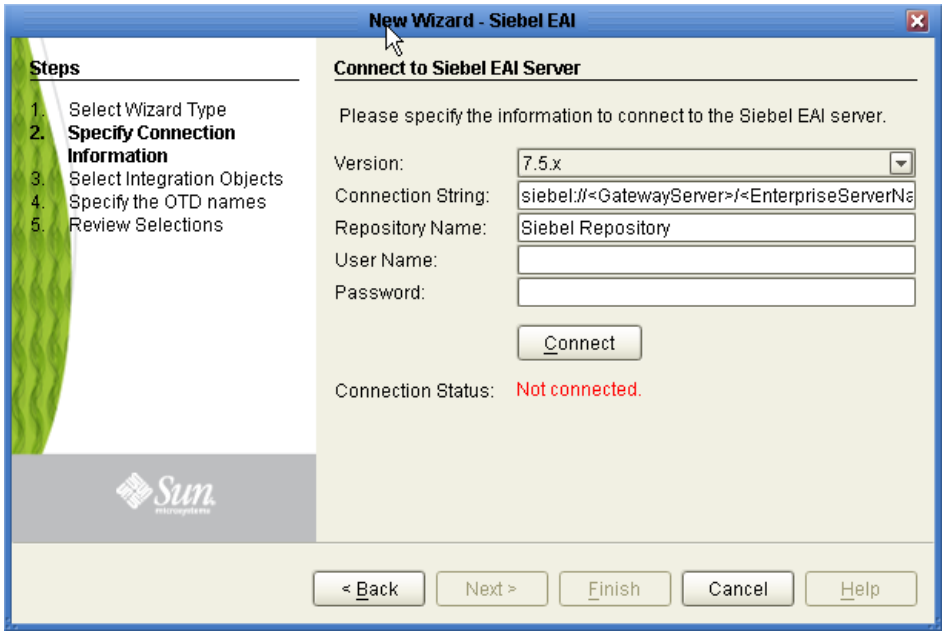
**Figure 24** Select Wizard Type window




### Specify Connection Information

1 From the **New Object Type Definition Wizard** window, select **SiebelEAIWizard** and click the **Next** button. The **Connect to Siebel EAI Server** window appears.

**Figure 25** Siebel Server Connection window

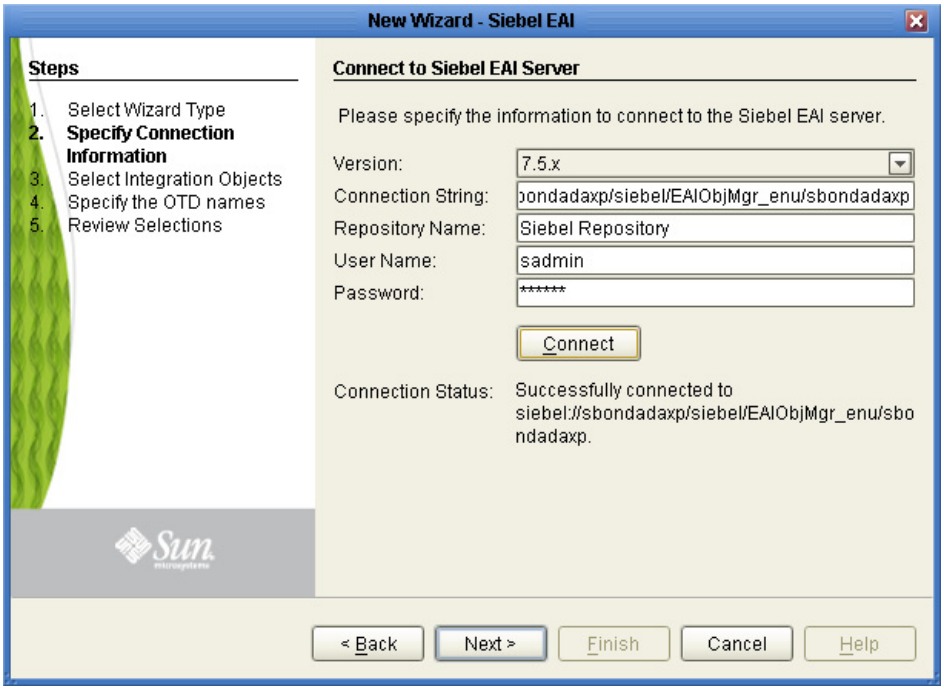



2 Enter the following information into the text fields:

- **Version:** the Siebel Server version (the same version used to create the OTD appears by default)
- Connection String:
  - **For Siebel 7.5.x:** `siebel://<GatewayServer>/<EnterpriseServerName>/<ApplicationObjectManager>/<SiebelServerName>`
  - **For Siebel 7.7 and 7.8.x:** `siebel://<SiebelServerName>:<port>/<EnterpriseServerName>/<ApplicationObjectManager>`
- Repository Name: Siebel Repository
- **User Name:** a valid user name
- **Password:** a valid password

3 Click the **Connect** button. A message appears confirming a successful connection.

**Figure 26** Successful Server Connection window

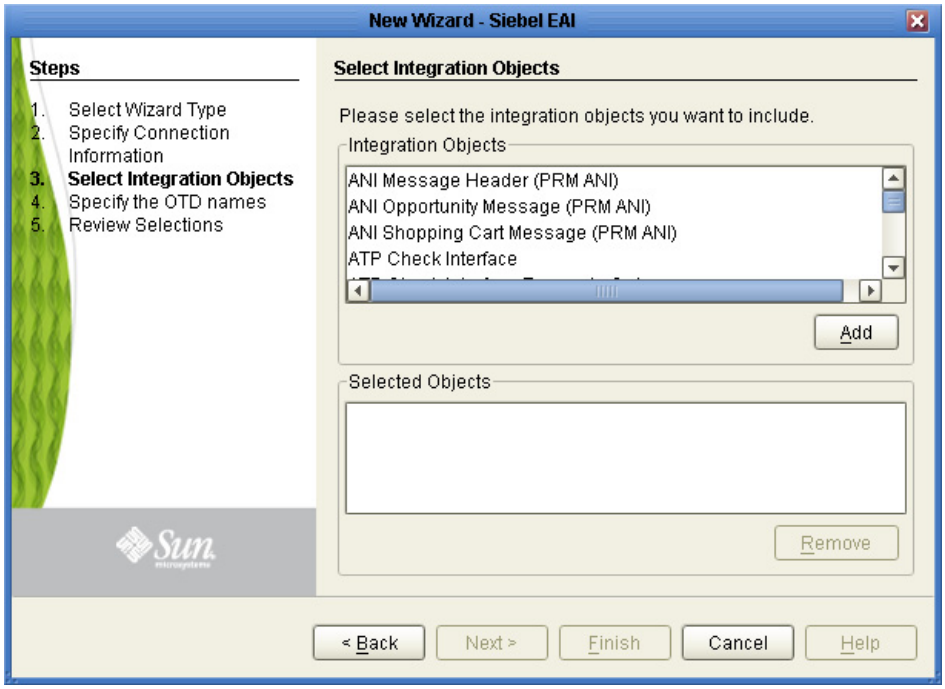



### Select Integration Objects

1 Click the **Next** button. The **Select Integration Objects** window appears. Scroll down the Integration Objects selection table and select **Account Interface**.

2 Click the **Add** button. Account Interface appears in the **Selected Objects** window.

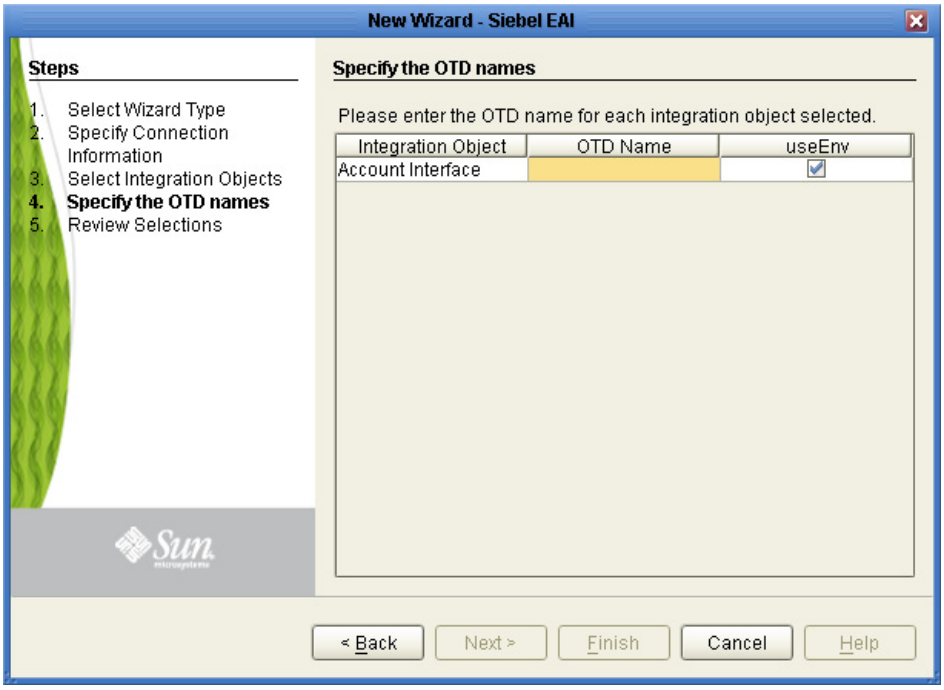
**Figure 27** Select Integration Objects window




### Specify OTD Names

1 Click the **Next** button. The **Specify the OTD names** window appears.

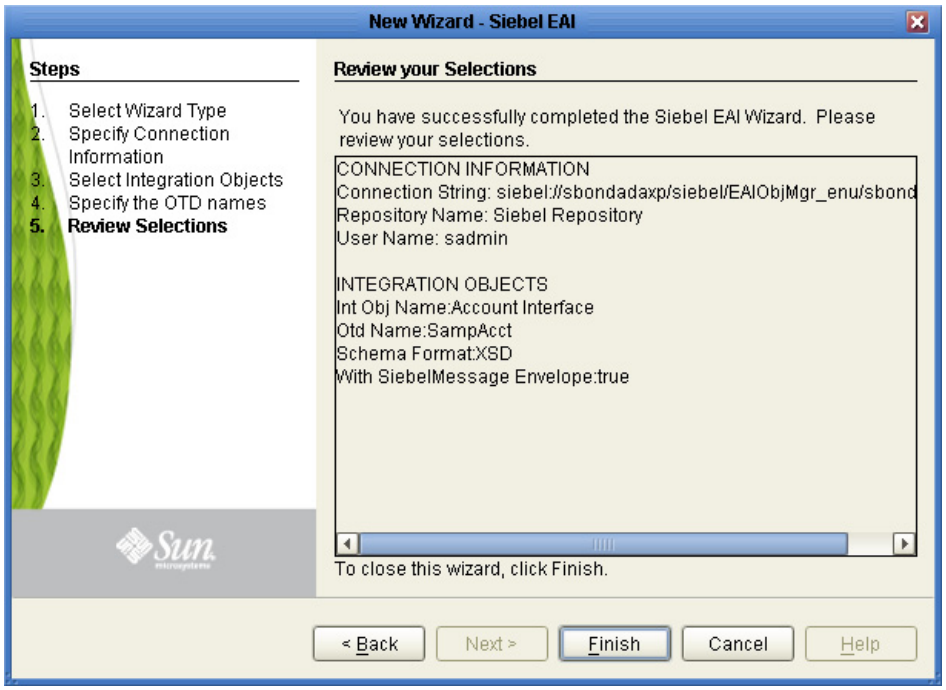
**Figure 28** Specify the OTD names window




2 In the **OTD Name** column, enter the name for the new OTD.

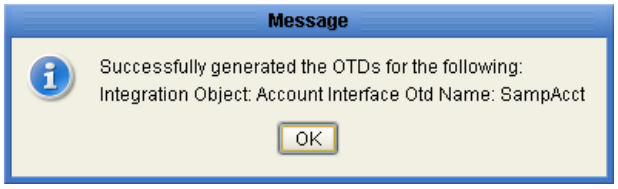
3 Click the **Next** button. The Review your Selections window appears.

**Figure 29** Review your Selections window




4 Click the **Finish** button. A message appears confirming the successfully generated OTD. See Figure 30.

**Figure 30** Completed OTD Message window




## 6.3.1 Relaunching the OTD

Steps required to relaunch an OTD include:

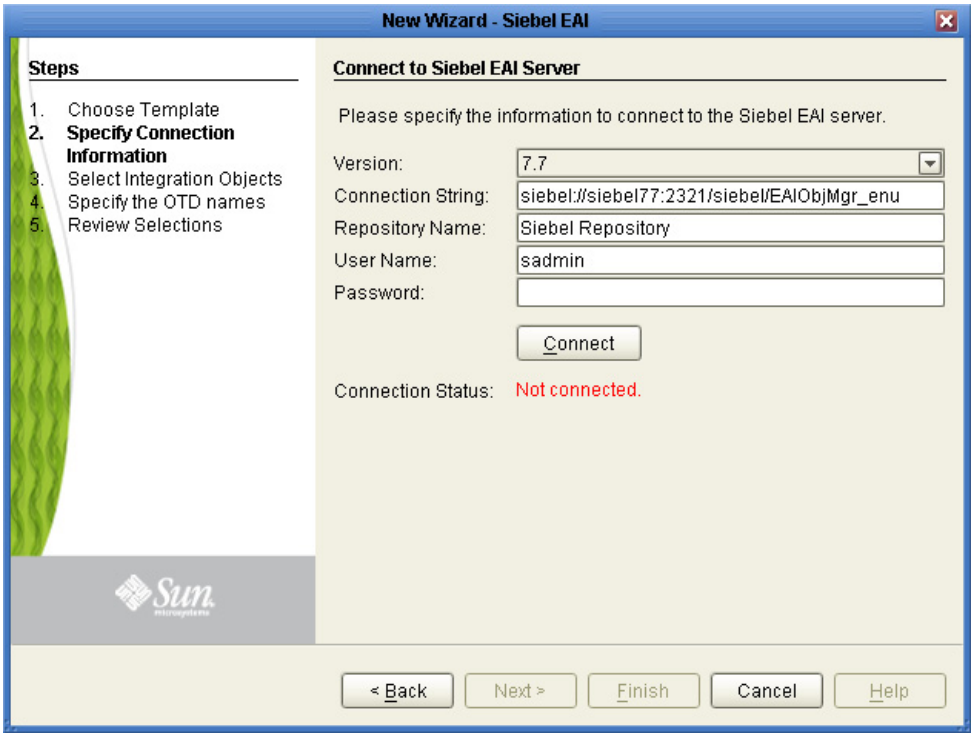
1 From the Project Explorer, right-click the OTD and select **Version Control > Check Out** from the shortcut menu. The Version Control - Check Out dialog box appears.

2 Select **Check Out on the Version Control - Check Out window.**

3 Right-click the OTD once again and select Relaunch.

The C**onnect to Siebel EAI Server** window appears.

**Figure 31** Siebel Server Connection window




The fields (with the exception of **Password**) will be populated with metadata information selected when building the original OTD.

4 Enter a valid password in the **Password** field.

5 Click the **Connect** button. A message appears confirming a successful connection.

6 Click the **Next** button. The **Select Integration Objects** window appears. The **Selected Objects** window will already be populated with the previously selected Integration Object.

**Important:** *If you attempt to select a different Integration Object, an error dialogue will appear.*

7 Click the **Next** button. The **OTD Name** column will already be populated with the original OTD's name.

8 Click the **Next** button. The Review your Selections window appears.

9 Click the **Finish** button. A message appears confirming the successfully generated OTD.

Chapter 7

# Implementing the Siebel EAI eWay Sample Projects

This chapter provides an introduction to the Siebel EAI eWay components, and information on how these components are created and implemented in a Sun Java Composite Application Platform Suite Project. Sample Projects are designed to provide an overview of the basic functionality of the Siebel EAI eWay by identifying how information is passed between Java CAPS and the Siebel EAI application.

It is assumed that you understand the basics of creating a Project using the Enterprise Designer. For more information on creating a Java CAPS Project, see the *eGate Tutorial* and the *eGate Integrator User's Guide*.

**What's in This Chapter**



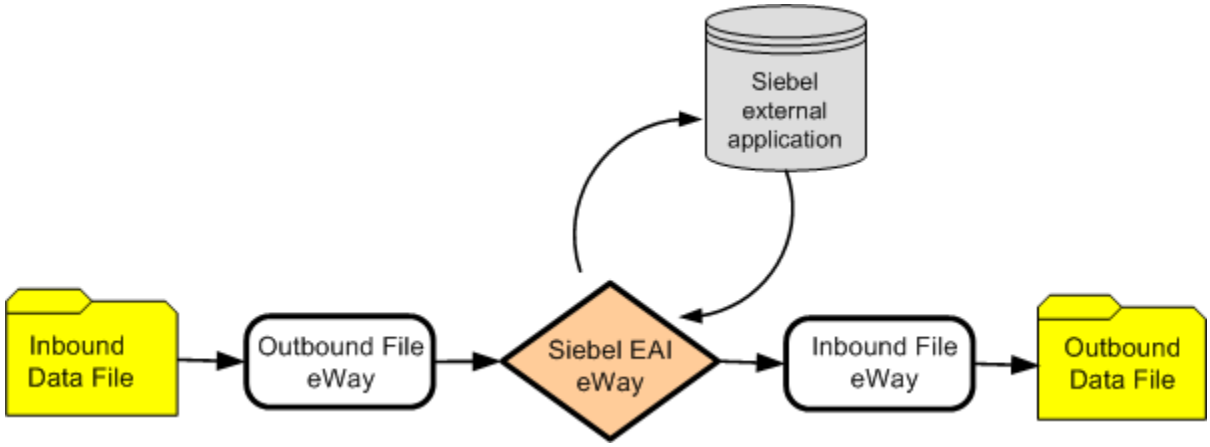
## 7.1 About the Siebel EAI Sample Projects

The Siebel EAI eWay **SiebelEAI_eWay_Sample.zip** file contains two sample Projects that provide basic instruction on using the Siebel EAI eWay in JCD, or the Business Process Excecution Language (BPEL) Projects.

Both the **SiebelEAI_bpUpsert** and **SiebelEAI_jcdUpsert** sample Projects contain an input file that accepts sample data and invokes a Siebel workflow to get data, and creates an output that writes the response code to a log file. Along the way, the Siebel Business process validates the data and enforces the necessary business rules. Both projects use the **SeeBeyond HTTP Update** Siebel workflow. Results are written out to an output file.

Figure 32 shows the business process used by the sample Projects.

**Figure 32** Siebel EAI Sample Project Data Flow




In addition to the sample Projects, the **SiebelEAI_eWay_Sample.zip** file also includes the sample input trigger file **SiebelEAI_sa.in.~in** and the sample output files **BPEL_sa_Upsert_output1.dat** (BPEL) and **JCD_sa_Upsert_output1.dat** (JCD).

## 7.2 Importing a Sample Project

Sample eWay Projects are included as part of the installation package. To import a sample eWay Project to the Enterprise Designer, do the following:

1. The sample files are uploaded with the eWay's documentation SAR file and downloaded from the Sun Composite Application Platform Suite Installer's Documentation tab. The **SiebelEAI_eWay_Sample.zip** file contains the various sample Project ZIP files and sample data. Extract the samples to a local file.
2. Save all unsaved work before importing a Project.
3. From the Enterprise Designer's Project Explorer pane, right-click the Repository and select **Import** from the shortcut menu. The **Import Manager** appears.
4. Browse to the directory that contains the sample Project ZIP file. Select the sample file and click **Import**. After the sample Project is succesfully imported, you can import additional samples or click **Close** to exit the Import Manager.
5. Before an imported sample Project can be run, you must do the following:
   - Create an Environment (see **"Creating an Environment" on page 71**)
   - Configure the eWays for your specific system (see **"Configuring the eWays" on page 72**)
   - Create a Deployment Profile (see **"Creating and Activating the Deployment Profile" on page 74**)
   - Create and start a domain (see **"Creating and Starting the Domain" on page 75**)

- Build and deploy the Project (see **"Building and Deploying the Project" on page 75**)

The following pages provide step-by-step instructions for creating the **SiebelEAI_bpUpsert** and **SiebelEAI_jcdUpsert** sample Projects.

## 7.3 Building, Deploying, and Running the SiebelEAI_bpUpsert Sample Project

The following provides step-by-step instructions for creating the **SiebelEAI_bpUpsert** sample Project.

Steps required to create the sample project include:



### 7.3.1 Creating a Project

The first step is to create a new Project in the Enterprise Designer.

1. Start the Enterprise Designer.
2. From the Project Explorer tree, right-click the Repository and select **New Project**. A new Project (**Project1**) appears on the Project Explorer tree.
3. Click twice on **Project1** and rename the Project (for this sample, **SiebelEAI_bpUpsert**).

### 7.3.2 Creating the OTD

The next step is to create a Object Type Definition (OTD) to interact with the Siebel EAI eWay.

**Steps required to create a Siebel EAI OTD include:**

1. Right-click your new Project in the Enterprise Designer's Project Explorer, and select **New** > **Object Type Definition**.

The New Object Type Definition Wizard window appears.

2 Select the **Siebel EAI OTD Wizard** from the list of OTD Wizards and click **Next**.

3 Enter the connection information for the Siebel application. Connection fields include:

- Version
- Connection String
- Repository Name
- User Name
- Password

4 Click **Connect**. A message appears confirming a successful connection.

5 Click the **Next** button. The **Select Integration Objects** window appears. Scroll down the Integration Objects selection table and select the **Sample Account** Integration Object.

6 Click the **Add** button. Sample Account appears in the Selected Objects window.

7 Click the **Next** button. The **Specify the OTD names** window appears.

8 In the OTD Name column, enter **otdSampleAccount** for the new OTD.

9 Click the **Next** button. The Review your Selections window appears.

10 Click the **Finish** button. A message appears confirming the successfully generated OTD.

## 7.3.3 Creating the Business Process

Steps required to create the Business Process include:

- Creating the business process flow
- Configuring the modeling elements

### Sample Project Business Process

The data used for the sample Projects are contained within an input file called **siebelEAI_sa.in**, which displays data in XML like tags. The file contents is below:

```
<ListOfSampleAccount>
    <Account>
        <System></System>
        <Culture></Culture>
        <CurrencyCode>USD</CurrencyCode>
        <Description></Description>
        <Division></Division>
        <HomePage></HomePage>
        <IntegrationId></IntegrationId>
        <LineofBusiness></LineofBusiness>
        <Location></Location>
        <Name>SampleAccount</Name>
        <ProjectName></ProjectName>
        <Type></Type>
        <ListOfAccount_Organization>
```

```
                <Account_Organization IsPrimaryMVG="Y">
                    <IntegrationId></IntegrationId>
                    <Organization>Default Organization</Organization>
                </Account_Organization>
            </ListOfAccount_Organization>
        </Account>
    </ListOfSampleAccount>
```

Creation of a business process includes:

- Dragging and dropping business process activities from the Project explorer tree to the eInsight Business Process Designer's modeling canvas.
- Connecting logical business activities together.
- Adding business rules between activities.

**Steps to create a business process flow include:**

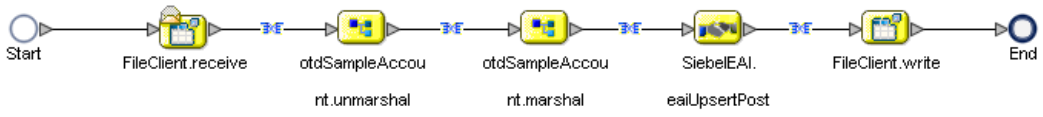
1. Right-click your new Project in the Enterprise Designer's Project Explorer, and select **New** > **Business Process** from the shortcut menu. The eInsight Business Process Designer appears and **BusinessProcess1** is added to the Project Explorer tree. Rename **BusinessProcess1** to **bp_sa_Upsert**.
2. Add the following Activities to the Business Process Designer canvas.

**Table 11** Business Process Activities

| Business Process | Activity |
|---|---|
| bpelPsInsert | ▪ FileClient.Receive<br>▪ otdSampleAccount.unmarshal<br>▪ otdSampleAccount.marshal<br>▪ SiebelEAI.eaiUpsertPost<br>▪ FileClient.Write |

Figure 33 provides an example of a completed business process.

**Figure 33** Example Business Process




## Adding Business Process Activities

An eInsight Business Process Activity can be associated with the Siebel EAI eWay during system design phase. To make this association, select the desired operators under eWay in the Enterprise Explorer and drag it onto the eInsight Business Process Designer canvas.

The **bp_sa_Upsert** business process has the following operations available:

- receive
- unmarshal
- marshal
- eaiPost
- write

The operation is automatically changed to an Activity with an icon identifying component that is the basis for the Activity. At run time, eInsight invokes each the order defined in the Business Process. Using eInsight's Web Services interface, Activity in turn invokes the Siebel EAI eWay.

## Available Web Service Operations

Siebel EAI contains several web service operations, located at **SeeBeyond** > **eWays** > **SiebelEAIeWay** > **Siebel EAI**.

Web service operations include:

- **eaiPost**—uses the *Sample Account* integration object with the **SeeBeyond HTTP Post** workflow.
- **eaiQueryPost**—uses the *Sample Account* integration object with the **SeeBeyond HTTP Query** workflow.
- **eaiUpsertPost**—uses *Sample Account* integration object with the **SeeBeyond HTTP Update** workflow.
- **eaiDeletePost**—uses *Sample Account* integration object with the **SeeBeyond HTTP Delete** workflow.
- **eaiwebportalReceive**—a web service operation that supports the inbound functionality in the business process.
- **eaiwebportalRequest**—a web service operation that supports the outbound functionality in the business process.

## Configuring the Modeling Elements

Business rules are defined and configured between the business process activities located on the modeling canvas. The sample Projects contain business rules between each of the activities listed in the business process flow.

**Note:** *Review the eInsight Business Process Manager User's Guide for a more detailed description of the steps required to connect and add business rules to a modeling elements in a business process.*

Sample eInsight Projects included on the installation CD-ROM contain similar business rules. The business process described below uses the **SiebelEAI_bpUpsert** sample as illustration.

During the business process, the sample Project:

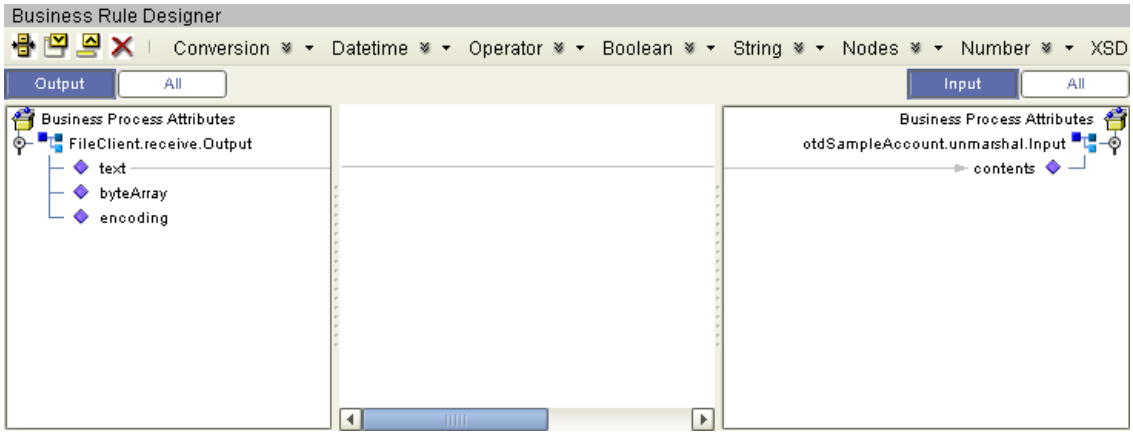
- Unmarshals data into an OTD

- Marshals data from the Siebel workflow
- Calls the PostSiebelform
- Verifies the Response Code from Siebel
- Writes the output log file

## Unmarshal Data

Data is first read as a long string from the input file, then broken up into recognizable fields and placed into the sample account table. Unmarshaling is illustrated by the the business rule between the **FileClient.write** Activity and **otdSampleAccount.unmarshal** Activity, as seen in Figure 34.

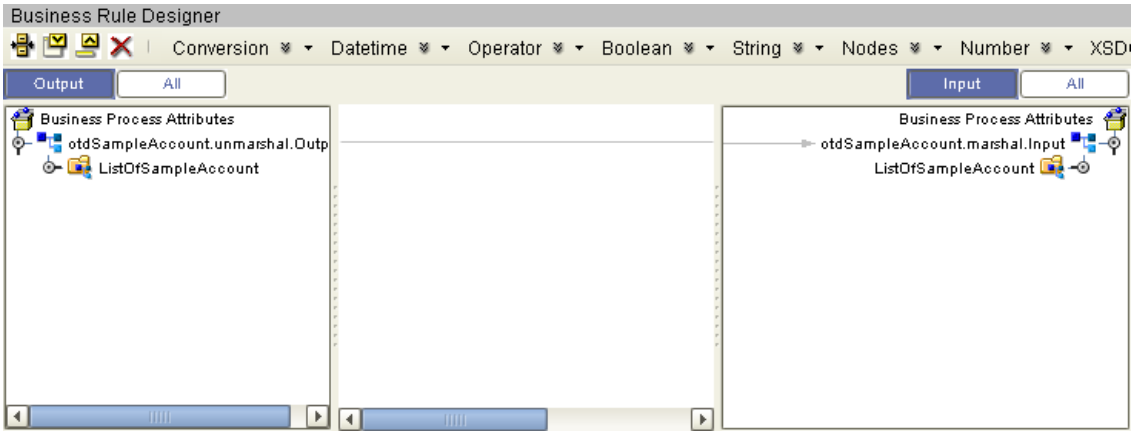
**Figure 34** Unmarshaling Data into the OTD




## Marshal Data into the Siebel Workflow

The OTD data is grabbed, marshaled to string format, and then assigned to the Sample Account table (otdSampleAccount.marshal.Input OTD). Marshaling data into the Siebel Workflow is illustrated by the the business rule between the **otdSampleAccount.unmarshal** Activity and **otdSampleAccount.marshal** Activity, as seen in Figure 35.

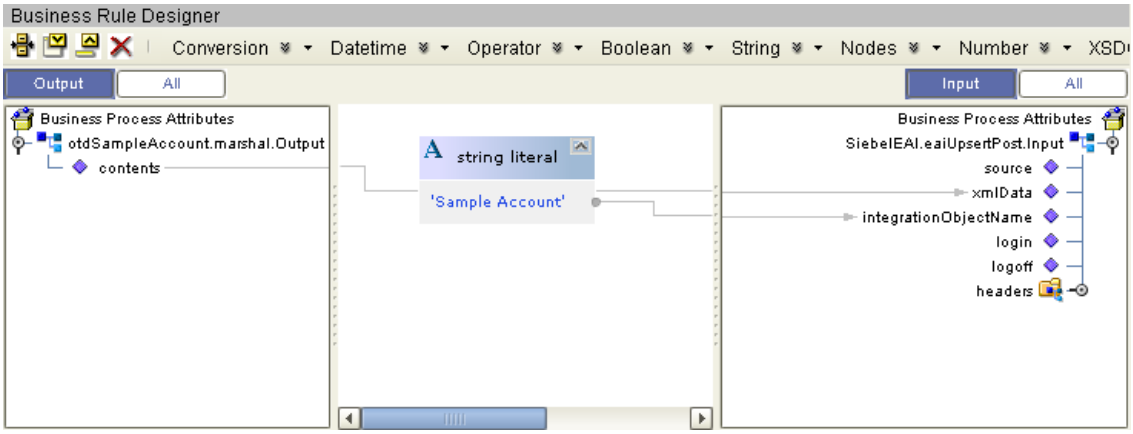
**Figure 35** Marshaling Data from the Siebel Workflow




## Call the PostSiebelform

The PostSiebelform is a method that invokes the **SeeBeyond HTTP Update** workflow. Data from the xmlData field is forwarded in an XML-formatted message to the Siebel Workflow that manages and enforces the Business Process. Calling the PostSiebelform is illustrated by the the business rule between the **otdSampleAccount.marshal** Activity and **SiebelEAI.eaiUpsertPost** Activity, as seen in Figure 36.

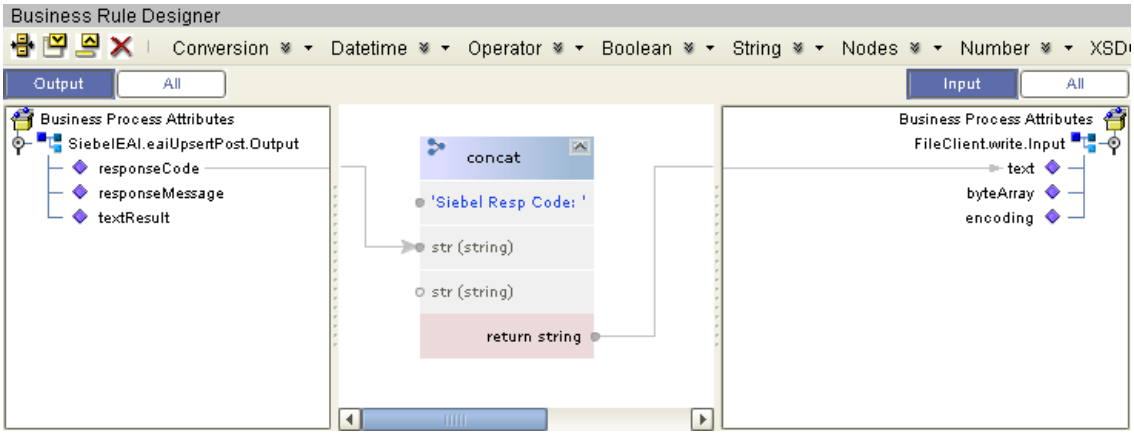
**Figure 36** Calling the PostSiebelform




## Verify the Response Code

During run-time, the Workflow invokes the Siebel EAI XML Converter, which converts the information from XML into the Siebel internal format. The responseCode responseMessage and textResult are concatenated into a single response back to the user. Verifying the response code is illustrated by the the business rule between the **SiebelEAI.eaiUpsertPost** Activity and **FileClient.write** Activity, as seen in Figure 37.

**Figure 37** Verifying the Response Code




### Write the Response Code

The final step is to write the response code log file, which verifies the operational status of the Workflow

## 7.3.4 Creating a Connectivity Map

A Connectivity Map provides a canvas for assembling and configuring a Project's components. The **SiebelEAI_bpUpsert** Project only uses one Connectivity Map.

### To create a Connectivity Map

1. From the Project Explorer tree, right-click the new **SiebelEAI_bpUpsert** Project and select **New > Connectivity Map** from the shortcut menu.
2. The new Connectivity Map appears and a node for the Connectivity Map is added under the Project on the Project Explorer tree labeled **CMap1**. Rename the Connectivity Map **cmbpUpsert**.

## Selecting External Applications

When creating a Connectivity Map, you can associate any Service, in this case a Business Process, with an external application. For example, to establish a connection to Siebel EAI, you must first select **Siebel EAI** as the external application to use in your Connectivity Map.

### To select external applications

1. Click the **External Application** icon on the Connectivity Map toolbar.
2. Select the external applications necessary for your Project. For this sample, select the **File** and **Siebel EAI** external applications. Icons representing these external applications are then added to the Connectivity Map toolbar.

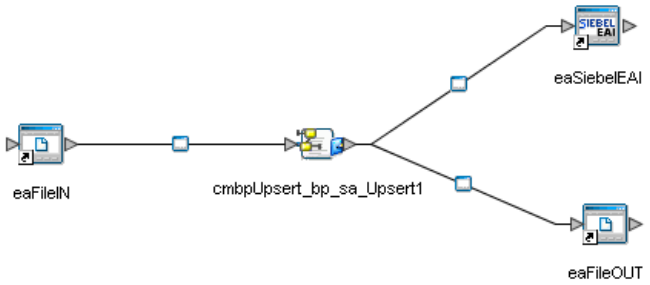
## Populating the Connectivity Map

Add the Project components to the **cmbpUpsert** Connectivity Map by dragging the icons from the toolbar to the canvas. For this sample Project, drag and drop the following components onto the Connectivity Map canvas.

- File External Application (2)
- Business Service (a service is a container for Java Collaborations, Business Processes, and so forth)
- Siebel EAI External Application

Figure 38 shows the components in the Connectivity Map.

**Figure 38** Connectivity Map With Components: SiebelEAI_bpUpsert




Rename the **Service1** component to **bp_sa_Upsert1**. Name the other components as shown in Figure 38. Be sure to save the new Connectivity Map before you proceed. You can click **Save** on the Enterprise Designer toolbar for this purpose.

## Defining the Business Process

Define your Business Process by combining the Business Process icon with the Service icon in the Connectivity Map. To do so, drag and drop the **bp_sa_Upsert** icon from the **Project Explorer** tree onto the Connectivity Map's **bp_sa_Upsert1** Service icon. If the operation is successfully defined, the gears on the **bp_sa_Upsert1** icon change from red to yellow.

## Binding the eWay Components

The final step in creating a Connectivity Map is binding the eWay components together.

**Steps required to bind eWay components together:**

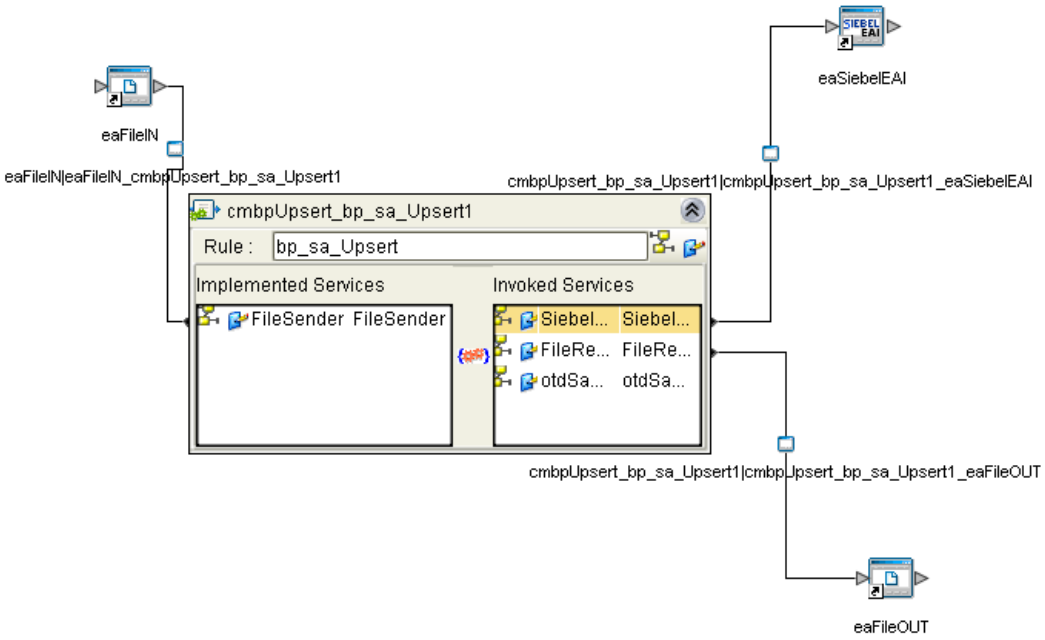
1. Open the **cmbpUpsert** Connectivity Map and double-click the **bp_sa_Upsert1** Business Process. The **bp_sa_Upsert1** Binding dialog box appears.
2. From the **bp_sa_Upsert1** Binding dialog box, map **FileSender** (under Implemented Services) to the **eaFileIN** (File) External Application. To do this, click on **FileSender** in the **bp_sa_Upsert1** Binding dialog box, and drag the cursor to the **eaFileIN**

External Application in the Connectivity Map. A link is now visible between **eaFileIN** and **bp_sa_Upsert1**.

3 From the **bp_sa_Upsert1** Binding dialog box, map **SiebelEAIReceiver** (under Invoked Services) to the **eaSiebelEAI** External Application.

4 From the **bp_sa_Upsert1** Binding dialog box, map **FileReceiver** to the **eaFileOUT** External Application, as seen in Figure 39.

**Figure 39** Connectivity Map - Associating (Binding) the Project's Components




## 7.3.5 Creating an Environment

Environments include the external systems, Logical Hosts, Integration Servers, and message servers used by a Project and contain the configuration information for these components. Environments are created using the Enterprise Designer's Environment Editor.

1 From the Enterprise Designer's Enterprise Explorer, click the **Environment Explorer** tab.

2 Right-click the Repository and select **New Environment**. A new Environment is added to the Environment Explorer tree.

3 Rename the new Environment to **envSiebelEAI_BPEL**.

4 Right-click **envSiebelEAI_BPEL** and select **New > File External System**. Name the External System **esFile** and click **OK**. **esFile** is added to the Environment Editor.

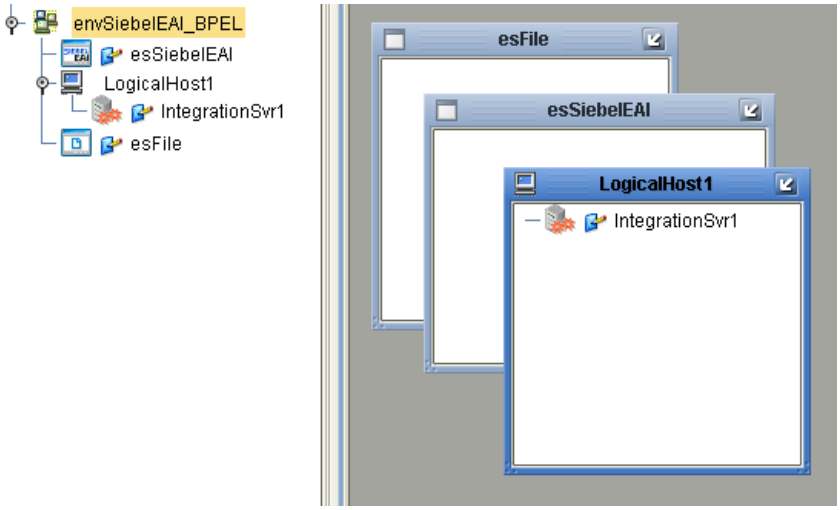
5 Right-click **envSiebelEAI_BPEL** and select **New > Siebel EAI External System**. Name the External System **esSiebelEAI** and click **OK**. **esSiebelEAI** is added to the Environment Editor.

6 Right-click **envSiebelEAI_BPEL** and select **New > Logical Host**. **LogicalHost1** is added to the Environment Editor.

7 From the Environment Explorer tree, right-click **LogicalHost1** and select **New > Sun SeeBeyond Integration Server**. A new Integration Server (**IntegrationSvr1**) is added to the Environment Explorer tree under LogicalHost1.

8 Save changes to the repository. The Environment Explorer and Environment Editor now appear as displayed in Figure 40.

**Figure 40** Environment Editor - envSiebelEAI_BPEL




9 Save your current changes to the Repository.

## 7.3.6 Configuring the eWays

eWays facilitate communication and movement of data between the external applications and the eGate system. Each Connectivity Map in the **SiebelEAI_bpUpsert** sample Project use three eWays that are represented as a nodes between the External Applications and the Business Process, as seen in Figure 38.

You must configure eWay properties in both the Connectivity Map and the Environment Explorer.

1 From the **cmbpUpsert** Connectivity Map, double-click the inbound **eaFileIN** eWay. The **Properties Editor** opens to the inbound File eWay properties.

2 Modify the properties for your system, including the settings for the inbound File eWay in Table 12, and click **OK**.

**Table 12** cmbpUpsert - Inbound File eWay Settings

| **FileClientIN eWay Connection Parameters** | |
|---|---|
| Input file name | SiebelEAI_sa.in.~in |

3 From the **cmbpUpsert** Connectivity Map, modify the outbound **eaFileOUT** eWay properties for your system, including the settings in Table 13.

**Table 13** cmbpUpsert - Outbound File eWay Settings

| Outbound File eWay Connection Parameters | |
|---|---|
| Output file name | BPEL_sa_Upsert_output1.dat |

4 From the **Environment Explorer** tree, right-click the File eWay External System (**esFile** in this sample), and select **Properties** from the shortcut menu. The Properties Editor appears.

5 Modify the File eWay Environment properties for your system, including the settings in Table 14, and click **OK**.

**Table 14** File eWay Environment Properties

| File eWay Environment Properties | |
|---|---|
| **Inbound File eWay > Parameter Settings** Set as directed, otherwise use the default settings | |
| Directory | *Select a directory, for example C:/DATA/input/* |
| **Outbound File eWay > Parameter Settings** Set as directed, otherwise use the default settings | |
| Directory | *Select a directory, for example C:/DATA/output* |

## Configuring the Siebel EAI Environment Properties

1 From the **Environment Explorer** tree, right-click the **esSiebelEAI** External System and select **Properties** from the shortcut menu. The Properties Editor appears.

2 Modify the Siebel EAI eWay Environment properties for your system, including the following settings:

- Siebel EAI
- HTTP Settings
- Proxy Configuration
- Security

For further information on configuring the Siebel EAI eWay, see **"eWay Environment Properties" on page 46**.

## Configuring the Integration Server

You must set your SeeBeyond Integration Server Password property before deploying your Project.

1 From the Environment Explorer, right-click **IntegrationSvr1** under your **Logical Host**, and select **Properties** from the shortcut menu. The Integration Server Properties Editor appears.

2 Click the **Password** property field under **Sun SeeBeyond Integration Server Configuration**. An ellipsis appears in the property field.

3 Click the ellipsis. The **Password Settings** dialog box appears.

4 Enter **STC** as the **Specific Value** and as the **Confirm Password**, and click **OK**.

5 Click **OK** to accept the new property and close the Properties Editor.

For more information on deploying a Project see the *Sun SeeBeyond Java™ Composite Application Platform Suite Deployment Guide*.

## 7.3.7 Creating and Activating the Deployment Profile

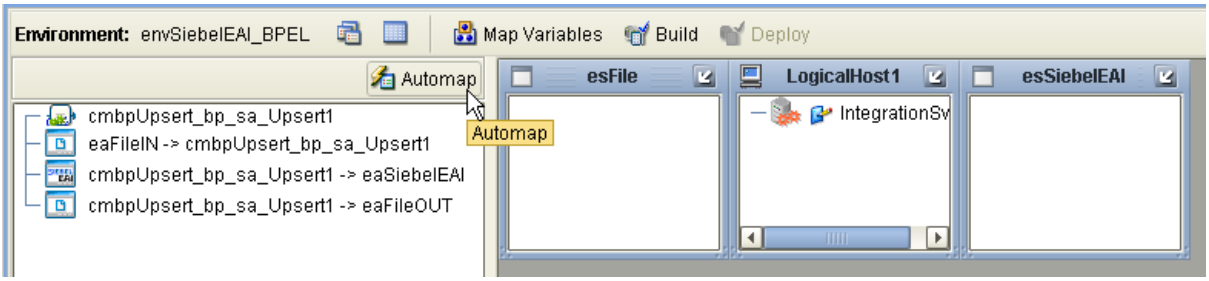
Deployment Profiles are used to assign Collaborations and message destinations to the Integration Server and message server. Deployment profiles are created using the Deployment Editor.

1 From the Project Explorer, right-click the **SiebelEAI_bpUpsert** Project and select **New** > **Deployment Profile**.

2 Enter a name for the Deployment Profile (for this sample **dpSiebelEAI_BPEL**). Select **envSiebelEAI_BPEL** as the Environment and click **OK**.

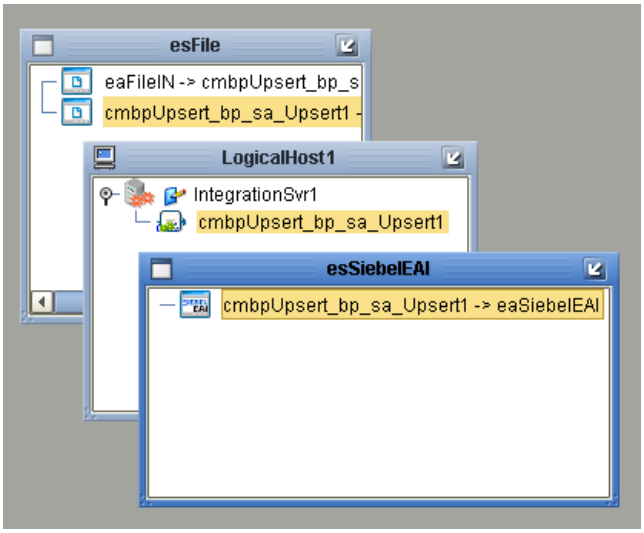
3 From the Deployment Editor toolbar, click the **Automap** icon (see Figure 41).

**Figure 41** dpSiebelEAI_BPEL Deployment Profile




The Project's components are automatically mapped to their system windows (see Figure 42).

**Figure 42** dpSiebelEAI_BPEL Deployment Profile Automapping




4 Save your current changes to the Repository.

## 7.3.8 Creating and Starting the Domain

To deploy your Project you must first create a domain. After the domain is created, the Project is built and then deployed.

### Create and Start the Domain

1. Navigate to your **<*JavaCAPS51*>\logicalhost** directory (where <*JavaCAPS51*> is the location of your Java Composite Application Platform Suite installation.
2. Double-click the **domainmgr.bat** file. The **Domain Manager** appears.
3. If you have already created a domain, select your domain in the Domain Manager and click the **Start an Existing Domain** button. Once your domain is started, a green check mark indicates that the domain is running.
4. If there are no existing domains, a dialog box indicates that you can create a domain now. Click **Yes**. The **Create Domain** dialog box appears.
5. Make any necessary changes to the **Create Domain** dialog box and click **Create**. The new domain is added to the Domain Manager. Select the domain and click the **Start an Existing Domain** button. Once your domain is started, a green check mark indicates that the domain is running.

## 7.3.9 Building and Deploying the Project

The Build process compiles and validates the Project's Java files and creates the Project EAR file.

### Build the Project

1. From the Deployment Editor toolbar, click the **Build** icon for each of your Deployment Profiles.
2. If there are any validation errors, a **Validation Errors** pane will appear at the bottom of the Deployment Editor and displays information regarding the errors. Make any necessary corrections and click **Build** again.
3. After the Build has succeeded you are ready to deploy your Project.

### Deploy the Project

1. From the Deployment Editor toolbar, click the **Deploy** icon. Click **Yes** when the **Deploy** prompt appears. Do this for both of your Deployment Profiles.
2. A message appears when the project is successfully deployed. You can now test your sample.

**Note:** *Projects can also be deployed from the Enterprise Manager. For more information about using the Enterprise Manager to deploy, monitor, and manage your projects, see the Sun SeeBeyond eGate™ Integrator System Administration Guide.*

## 7.3.10 Running the Sample

The **SiebelEAI_bpUpsert** Project includes the following sample files:

- **SiebelEAI_sa.in.~in** (input file)
- **BPEL_sa_Upsert_output1.dat** (sample output file example)

To run your deployed sample Project, do the following:

1. From your configured input directory, paste (or rename) the sample input file to trigger the eWay.
2. From your output directory, verify the output data.

## 7.4 Building, Deploying, and Running the SiebelEAI_jcdUpsert Sample Project

The following provides step-by-step instructions for creating the **SiebelEAI_jcdUpsert** sample Project.

Steps required to create the sample project include:



### 7.4.1 Creating a Project

The first step is to create a new Project in the Enterprise Designer.

1. Start the Enterprise Designer.
2. From the Project Explorer tree, right-click the Repository and select **New Project**. A new Project (**Project1**) appears on the Project Explorer tree.
3. Right-click **Project1** and select **Rename** form the shortcut menu. Rename the Project (for this sample, **SiebelEAI_jcdUpsert**).

### 7.4.2 Creating the OTD

The next step is to create a Object Type Definition (OTD) interact with the Siebel EAI eWay.

Follow the steps outlined in **"Creating the OTD" on page 63** to create a new OTD using a **Sample Account** Integration Object. Name the new OTD **otdSampleAccount**.

## 7.4.3 Creating the Collaboration Definition (Java)

The eGate Enterprise Designer contains a Collaboration Definition wizard (Java) that allows you to create Java-based Collaborations. You must use the wizard to create a Collaboration Definition before implementing the Collaboration.

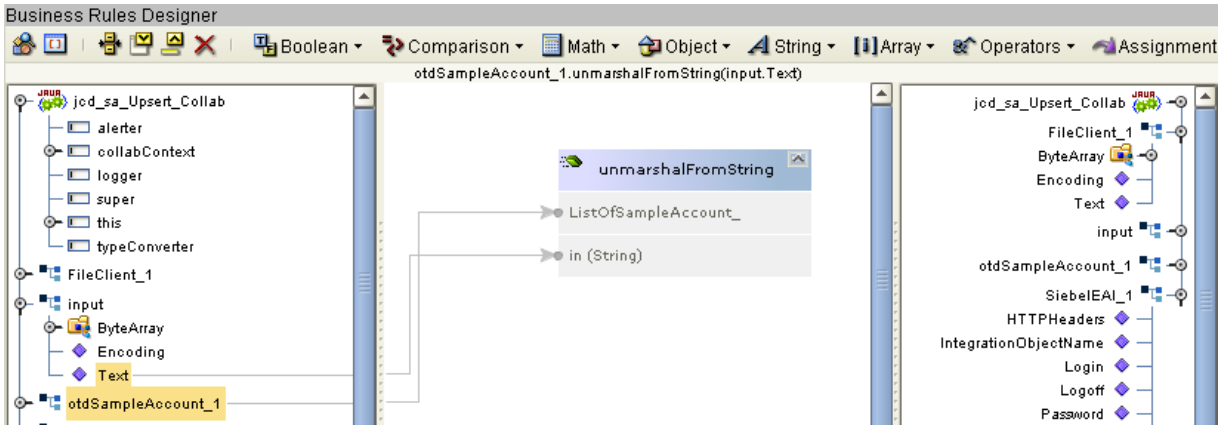
The Collaboration Editor user interface allows you to create the Business Rules that implement your business logic for a Java-based Collaboration. You can create the desired Business Rules for your Project by dragging and dropping values from a source OTD onto the nodes of a destination Siebel EAI OTD. Siebel EAI OTD nodes represent Siebel EAI functions, which are in turn able to call Siebel EAI eWay methods.

The Business Rules for the **jcd_sa_Upsert_Collab** Java Collaboration Definition are displayed in Figure 43.

**Figure 43** jcd_sa_Upsert_Collab Collaboration Definition

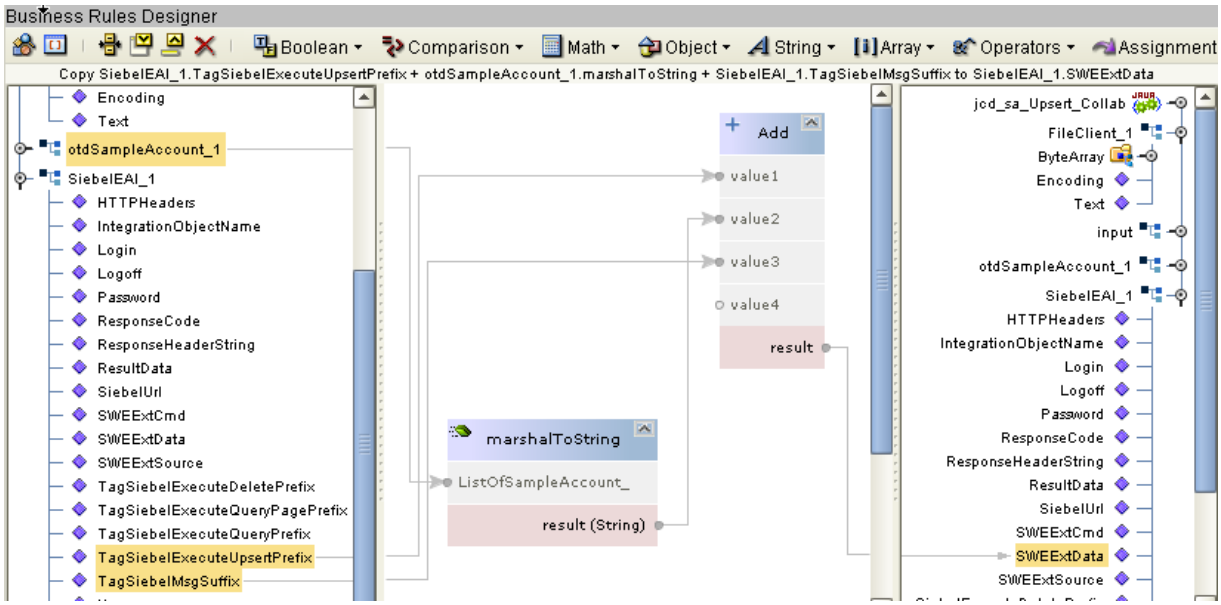

The **otdSampleAccount_1.unmarshalFromString(input.Text)** Business Rule is displayed in Figure 44.

**Figure 44** jcd_sa_Upsert_Collab Business Rule 1



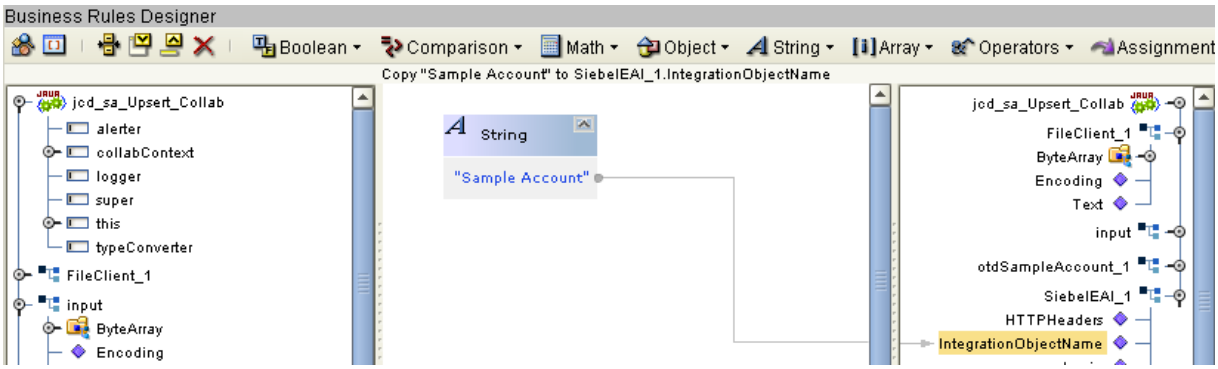

The **Copy SiebelEAI_1.TagSiebelExecuteUpsertPrefeix + otdSampleAccount_1.marshalToString + SiebelEAI_1.TagSiebelMsgSuffix to SiebelEAI_1.SWEExtData** Business Rule is displayed in Figure 45.

**Figure 45** jcd_sa_Upsert_Collab Business Rule 2



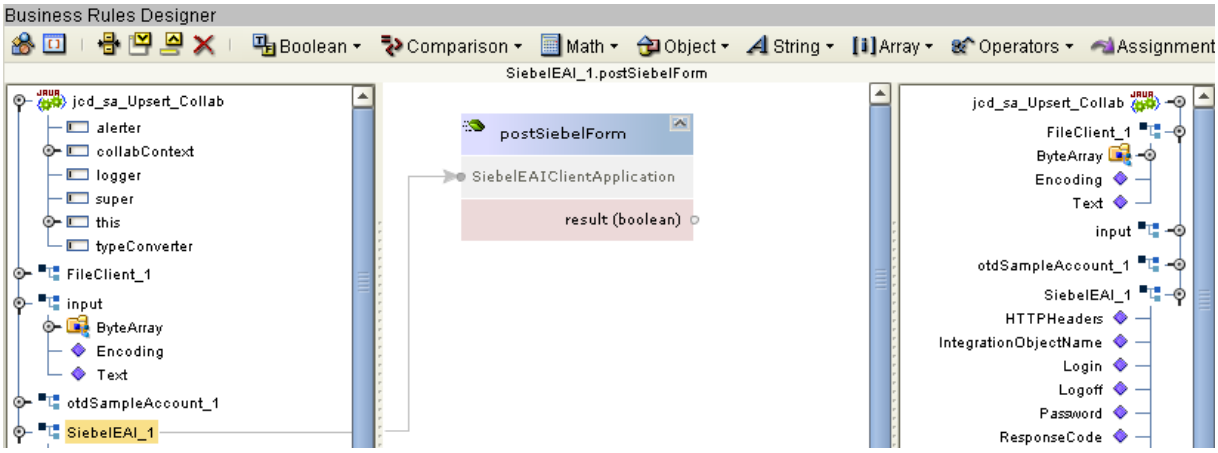

The **Copy "Sample Account" to SiebelEAI_1.IntegrationObjectName** Business Rule is displayed in Figure 46.

**Figure 46** jcd_sa_Upsert_Collab Business Rule 3



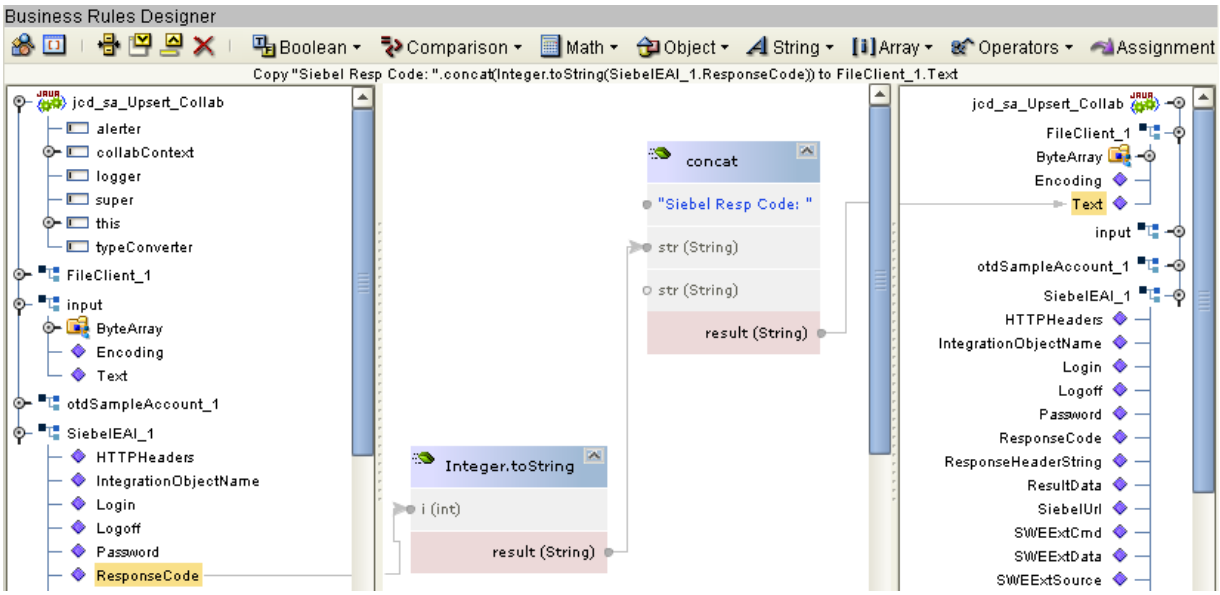

The **SiebelEAI_1.postSiebelForm** Business Rule is displayed in Figure 47.

**Figure 47** jcd_sa_Upsert_Collab Business Rule 4



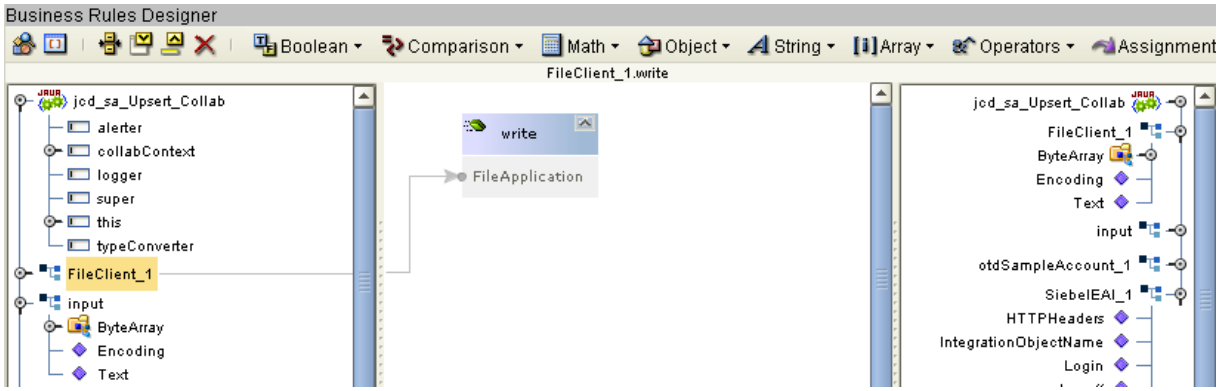

The **Copy "Siebel Resp Code:".concat(Integer.toString(SiebelEAI_1.ResponseCode)) to FileClient_1.Text** Business Rule is displayed in Figure 48.

**Figure 48** jcd_sa_Upsert_Collab Business Rule 5



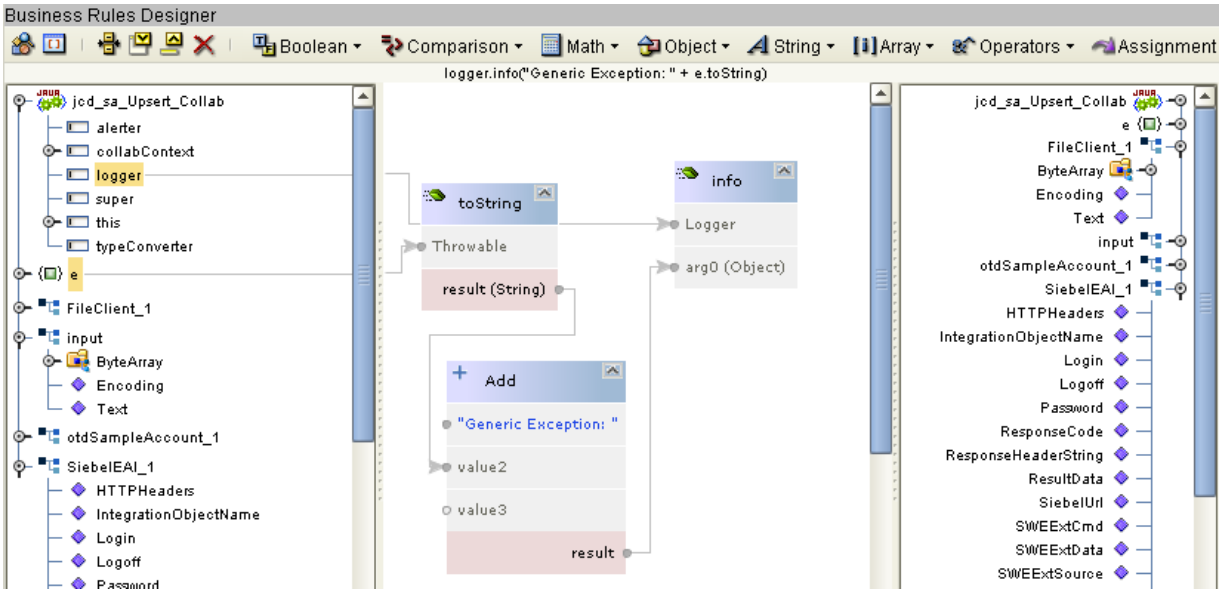

The **FileClient_1.write** Business Rule is displayed in Figure 49.

**Figure 49** jcd_sa_Upsert_Collab Business Rule 6




The **logger.info("Generic Exception:" + e.toString)** Business Rule is displayed in Figure 50.

**Figure 50** jcd_sa_Upsert_Collab Business Rule 7




## 7.4.4 Creating a Connectivity Map

A Connectivity Map provides a canvas for assembling and configuring a Project's components. The **SiebelEAI_jcdUpsert** Project only uses one Connectivity Map.

Follow the steps outlined in **"Creating a Connectivity Map" on page 69** to create a Connectivity Map. Name the Connectivity Map **cmjcdUpsert**.

### Selecting External Applications

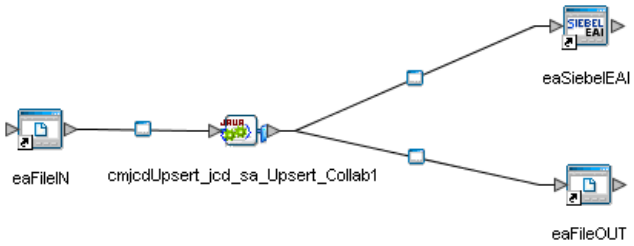
Follow the steps outlined in **"Selecting External Applications" on page 69** to select the external applications for the **SiebelEAI_jcdUpsert** Project's Connectivity Map.

### Populating the Connectivity Map

Follow the steps outlined in **"Populating the Connectivity Map" on page 70** to populate the Connectivity Map for the **SiebelEAI_jcdUpsert** Project's Connectivity Map.

Figure 38 shows the components in the Connectivity Map.

**Figure 51** Connectivity Map With Components: SiebelEAI_jcdUpsert




Rename the **Service1** component to **jcd_sa_Upsert1**. Name the other components as shown in Figure 51. Be sure to save the new Connectivity Map before you proceed. You can click **Save** on the Enterprise Designer toolbar for this purpose.

## Defining the Business Process

Define your Business Process by combining the Business Process icon with the Service icon in the Connectivity Map. To do so, drag and drop the **SiebelEAI_jcdUpsert** icon from the **Project Explorer** tree onto the Connectivity Map's **jcd_sa_Upsert_Collab** Service icon. If the operation is successfully defined, the gears on the **jcd_sa_Upsert_Collab** icon change from red to green.

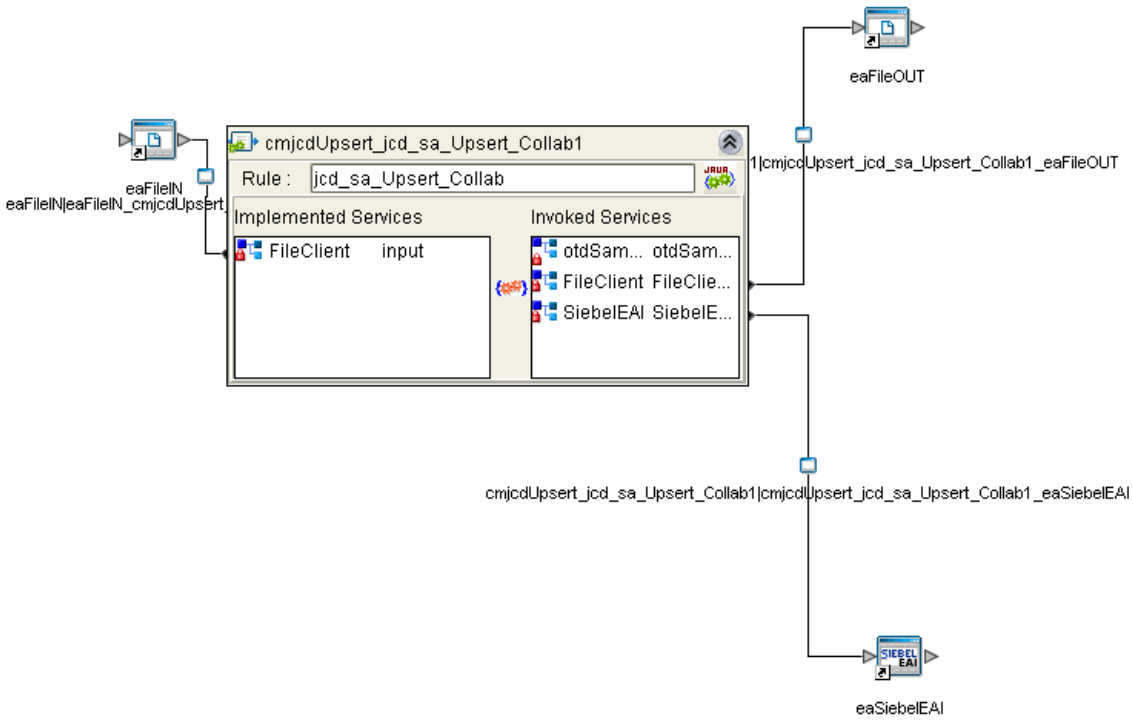
## Binding the eWay Components

The final step in creating a Connectivity Map is binding the eWay components together.

**Steps required to bind eWay components together:**

1. Open the **cmjcdUpsert** Connectivity Map and double-click the **jcd_sa_Upsert_Collab** Business Process. The **jcd_sa_Upsert_Collab** Binding dialog box appears.
2. From the **jcd_sa_Upsert_Collab** Binding dialog box, map **FileClient** (under Implemented Services) to the **eaFileIN** (File) External Application. To do this, click on **FileClient** in the **jcd_sa_Upsert_Collab** Binding dialog box, and drag the cursor to the **eaFileIN** External Application in the Connectivity Map. A link is now visible between **eaFileIN** and **jcd_sa_Upsert_Collab**.
3. From the **jcd_sa_Upsert_Collab** Binding dialog box, map **SiebelEAI_1** (under Invoked Services) to the **eaSiebelEAI** External Application.
4. From the **jcd_sa_Upsert_Collab** Binding dialog box, map **FileClient_1** to the **eaFileOUT** External Application, as seen in Figure 52.

**Figure 52** Connectivity Map - Associating (Binding) the Project's Components




## 7.4.5 Creating an Environment

Environments include the external systems, Logical Hosts, Integration Servers, and message servers used by a Project and contain the configuration information for these components. Environments are created using the Enterprise Designer's Environment Editor.

Follow the steps outlined in **"Creating an Environment" on page 71** to create an Environment for the **SiebelEAI_jcdUpsert** Project.

## 7.4.6 Configuring the eWays

eWays facilitate communication and movement of data between the external applications and the eGate system. Each Connectivity Map in the **SiebelEAI_jcdUpsert** sample Project use three eWays that are represented as a nodes between the External Applications and the Business Process.

You must configure eWay properties in both the Connectivity Map and the Environment Explorer.

1. From the **cmjcdUpsert** Connectivity Map, double-click the inbound **eaFileIN** eWay. The **Properties Editor** opens to the inbound File eWay properties.
2. Modify the properties for your system, including the settings for the inbound File eWay in Table 15, and click **OK**.

**Table 15** cmbpUpsert - Inbound File eWay Settings

| FileClientIN eWay Connection Parameters | |
|---|---|
| Input file name | SiebelEAI_sa.in.~in |

3 From the **cmjcdUpsert** Connectivity Map, modify the outbound **eaFileOUT** eWay properties for your system, including the settings in Table 16.

**Table 16** cmbpUpsert - Outbound File eWay Settings

| Outbound File eWay Connection Parameters | |
|---|---|
| Output file name | JCD_sa_Upsert_output1.dat |

4 From the **Environment Explorer** tree, right-click the File eWay External System (**esFile** in this sample), and select **Properties** from the shortcut menu. The Properties Editor appears.

5 Modify the File eWay Environment properties for your system, including the settings in Table 14, and click **OK**.

**Table 17** File eWay Environment Properties

| File eWay Environment Properties | |
|---|---|
| **Inbound File eWay > Parameter Settings** Set as directed, otherwise use the default settings | |
| Directory | *Select a directory, for example C:/DATA/input/* |
| **Outbound File eWay > Parameter Settings** Set as directed, otherwise use the default settings | |
| Directory | *Select a directory, for example C:/DATA/output* |

Follow the steps outlined in **"eWay Environment Properties" on page 46** to configure the eWay environment properties for the **SiebelEAI_jcdUpsert** Project.

## 7.4.7 Creating and Activating the Deployment Profile

Deployment Profiles are used to assign Collaborations and message destinations to the Integration Server and message server. Deployment profiles are created using the Deployment Editor.

Follow the steps outlined in **"Creating and Activating the Deployment Profile" on page 74** to create and deploy a deployment profile for the **SiebelEAI_jcdUpsert** Project.

## 7.4.8 Creating and Starting the Domain

To deploy your Project you must first create a domain. After the domain is created, the Project is built and then deployed.

Follow the steps outlined in **"Creating and Starting the Domain" on page 75** to create and deploy a domain for the **SiebelEAI_jcdUpsert** Project.

## 7.4.9 Building and Deploying the Project

The Build process compiles and validates the Project's Java files and creates the Project EAR file. Follow the steps outlined in **"Building and Deploying the Project" on page 75** to build and deploy the **SiebelEAI_jcdUpsert** Project.

## 7.4.10 Running the Sample

The **SiebelEAI_jcdUpsert** Project includes the following sample files:

- **SiebelEAI_sa.in.~in** (input file)
- **JCD_sa_Upsert_output1.dat** (sample output file example)

To run your deployed sample Project, do the following:

1. From your configured input directory, paste (or rename) the sample input file to trigger the eWay.
2. From your output directory, verify the output data.

# Index