SUN SEEBEYOND

eWAY™ e-MAIL ADAPTER USER'S GUIDE

Release 5.1.3

Part Number: 820-0986

Version 20070410083749

Contents

Chapter 1

Introduction

This chapter provides a brief overview of the e-Mail eWay and presents an outline of the information provided in this user's guide.

What's in This Chapter

1.1 About the e-Mail eWay Intelligent Adapter

The e-Mail eWay Intelligent Adapter (referred to as the e-Mail eWay throughout this document) enables the eGate system to exchange data with an SMTP (outbound) or POP3 (inbound) mail server. The eWay uploads messages to, and download messages from a mail server.

The e-Mail eWay enables many typically manual e-mail operations to be automated. Functions are provided to log into a server, create e-Mails, and add recipients, subject headers, content, and attachments. Functions are also provided to read data associated with an incoming message and to save attachments.

Collaborations can be created to intelligently send e-mails with formatted content, and to receive, parse and act upon incoming messages. The SSL feature is supported through the use of Java Secure Socket Extension (JSSE) version 1.0.3.

The e-Mail eWay's SSL (Secure Sockets Layer) feature provides secure communication channels for data exchanges, safe from unauthorized interception.

How does the eWay connect?

The e-Mail eWay takes advantage of widely used standard protocol.

- POP3 (Post Office Protocol) which can be thought of as a store and forward service. POP3 is used for retrieving e-mail from a mail account. POP3 requires the mail server name, the TCP/IP Port, the e-mail account name, and the e-mail account password.

- SMTP (Simple Mail Transfer Protocol) is used for sending e-mail to an account. SMTP, like POP3 requires the mail server name, the TCP/IP Port, the e-mail account name, and the e-mail account password. Some mail servers also require POP3 authentication to send messages.
- MIME (Multi-purpose Internet Mail Extensions) protocol. Servers insert the MIME header at the beginning of any Web transmission. Clients use the header to select appropriate applications for the type of data the header indicates. Multipart MIME messages (alternate text/HTML) and multiple attachments (other than nested MIME objects) are supported.

The eWay provides a custom Object Type Definition OTD (MailClient) for managing e-mail content, and for sending and retrieving e-mail. The OTD allows dynamic configuration of the connection fields within a Collaboration (allowing properties to be changed "on the fly" from within an existing Collaboration).

1.1.1 Japanese e-Mail Text Support

The e-Mail eWay supports Japanese character encoding in both the address and subject headers and text content of the e-mail message (both text/plain and text/HTML). The e-Mail eWay conforms to RFC2047 standards for Multipurpose Internet Mail Extensions (MIME) (see **"Enabling Japanese Character Support" on page 78**).

1.2 What's New in This Release

The Sun SeeBeyond eWay e-Mail Adapter includes the following changes and new features:

New for Version 5.1.3

- This is a maintenance release. No new features.

New for Version 5.1.2

- Supports automatic deployment of EAR files to WebLogic Application Server version 9.1.

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 for Runtime LDAP Configuration: eWay configuration properties now support LDAP key values.

- MDB Pool Size Support: Provides greater flow control (throttling) by specifying the maximum and minimum MDB pool size.
- 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 Administrator Guide*.

1.3 What's in This Document

This guide includes the following chapters:

- **Chapter 1 "Introduction"** provides an overview of the e-Mail eWay Intelligent Adapter.
- **Chapter 2 "Installing the e-Mail eWay"** provides the supported operating systems and system requirements for the e-Mail eWay. It also includes directions for installing the e-Mail eWay and accessing the accompanying documentation and sample Projects.
- **Chapter 3 "Configuring the e-Mail eWay Properties"** describes the process of configuring the e-Mail eWay to run in your environment.
- **Chapter 4 "Using the e-Mail eWay with eInsight"** describes how to use the e-Mail eWay with the Sun Java Composite Application Platform Suite's eInsight Business Process Manager and the Web Services interface.
- **Chapter 5 "Using the e-Mail eWay with Java Collaborations"** describes the features and functionality of the e-Mail eWay using the eGate Integrator and the Collaboration Editor (Java).
- **Chapter 6 "Using SSL"** describes the operation of the eWay's Secure Sockets Layer (SSL) feature.

1.3.1 e-Mail eWay Javadoc

The e-Mail eWay Javadoc documents the available Java methods provided with the e-Mail eWay. The Javadoc is uploaded with the eWay's documentation file, **EmaileWayDocs.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.

1.3.2 Scope of the Document

This user's guide provides a description of the e-Mail eWay Intelligent 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 e-Mail eWay Java methods, see the associated Javadoc.

1.3.3 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.

1.3.4 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**. ▪ On the **File** menu, click **Exit**. ▪ Select the **eGate.sar** file.
`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 10
Blue underlined	Hypertext links for Web addresses (URLs) or email addresses	http://www.sun.com

1.4 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

1.5 Documentation Feedback

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

CAPS_docsfeedback@sun.com

Chapter 2

Installing the e-Mail eWay

This chapter describes the system requirements and installation procedures for the e-Mail eWay.

What's in This Chapter

2.1 e-Mail eWay System Requirements

The e-Mail eWay Readme contains the latest information on:

- Supported Operating Systems
- System Requirements
- External System Requirements
- The e-Mail eWay Readme is uploaded with the eWay's documentation file (EmaileWayDocs.sar) and can be accessed from the Documentation tab of the Sun Java Integrator Suite Installer. Refer to the e-Mail eWay Readme for the latest requirements before installing the e-Mail eWay.

2.2 Installing the e-Mail 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 eWay on a JavaCAPS Supported System

Follow the directions for installing the Sun Java Composite Application Platform Suite in the *Sun Java Composite Application Platform Suite Installation Guide. After you have installed eGate or eInsight, do the following:*

1. From the Sun Java Composite Application Platform Suite Installer's **Select Sun Java Composite Application Platform Suite Products to Install** table (Administration tab), expand the **eWay** option.
2. 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)
   - **EmaileWay**

   To upload the e-Mail eWay User's Guide, Help file, Javadoc, Readme, and sample Projects, select the following:
   - **EmaileWayDocs**
3. Once you have selected all of your products, click **Next** in the top-right or bottom-right corner of the **Select Java Integration 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**. Your next selected product appears. Follow this procedure for each of your selected products. The **Installation Status** window appears and installation begins after the last SAR file has been selected.
5. Once your product's installation is finished, continue installing the Sun Java Composite Application Platform Suite as instructed in the *Sun Java Composite Application Platform Suite Installation Guide.*

Adding the eWay to an Existing Suite Installation

If you are adding the eWay to an existing Sun Java Composite Application Platform Suite installation, do the following:

1. Complete steps 1 through 4 above.
2. Once your product's installation is finished, 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.

2.3 Download Additional e-Mail eWay - SSL Runtime File

If you are using the e-Mail eWay with SSL, you must download and add an SSL Runtime file, (stcemailadapterappconnSSL.jar - SSL socket class) from the Sun Java Composite Application Platform Suite Installer.

To downloaded and apply this file, do the following:

1. From the Sun Java Composite Application Platform Suite Installer, click the **DOWNLOADS** tab.
2. From the **List of Components to Download**, double-click **Email eWay SSL Runtime**. The File Download dialog box appears. Click **Save**, and save the file to your desktop. The **stcemailadapterappconnSSL.jar** file appears on your desktop.
3. Copy and paste **stcemailadapterappconnSSL.jar** from your desktop, to the following location:

   `<JavaCAPS51>\logicalhost\is\lib`

   where <*JavaCAPS51*> is the directory in which the Sun Java Composite Application Platform Suite is installed.

2.4 Installing eWay Enterprise Manager plug-ins

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

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

1. From the Enterprise Manager:
   - A From the **Enterprise Manager**'s Explorer toolbar, click the **Configuration** icon.
   - B Click the **Web Applications Manager** tab, go to the **Auto-Install from Repository** tab, and connect to your Repository.
   - C Select the application plug-ins you require, and click **Install**. The application plug-ins are installed and deployed.
2. From the **Sun Java Composite Application Platform Suite Installer:**

A 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.

B Log onto the **Sun SeeBeyond Enterprise Manager**. From the **Enterprise Manager**'s Explorer toolbar, click the **Configuration** icon.

C Click the **Web Applications Manager** tab and the **Manage Applications** tab.

D Browse for and select the WAR file for the application plug-in that you downloaded, and click **Deploy**. The plug-in is installed and deployed.

e-Mail eWay Alert Codes

You can view and delete 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 alert codes are displayed under the **Results** section. If your eWay alert codes are not available 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 available this eWay's alert codes is displayed in Table 2.

Table 2 e-Mail eWay Alert Codes

Alert Code	Description	User Action
EMAILEWAY-CHECKEMAIL-FAILED000004	Failed to check for available e-mail message(s); host is {0}, using {1} port {2}.	Some component of the e-Mail is not supported by the e-Mail eWay. e-Mail message could not be parsed. Refer to the log for more information.
EMAILEWAY-CONNECT-FAILED000001	Failed to connect to host {0} on port {1} as user {2}.	▪ e-Mail server is not available. Verify that the e-Mail server is running and that you are able to connect to the server. ▪ Parameters are not configured properly. Verify that your e-Mail eWay property values are configured correctly. Refer to the log for more information.

Alert Code	Description	User Action
EMAILEWAY-RECV-FAILED000003	Failed to receive e-mail message; host is {0}, using {1} port {2}.	Parameters are not configured properly. Verify that your e-Mail eWay property values are configured correctly. Refer to the log for more information.
EMAILEWAY-SEND-FAILED000002	Failed to send e-mail message; host is {0}, using smtp port {1}.	Parameters are not configured properly. Verify that your e-Mail eWay property values are configured correctly. Refer to the log for more information.

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.*

2.4.1 After Installation

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

2.5 ICAN 5.0 Project Migration Procedures

This section describes how to transfer your current ICAN 5.0 Projects to Sun Java Composite Application Platform Suite, version 5.1.3. Only Projects developed on ICAN 5.0.2 and above can be migrated successfully to the Sun Java Composite Application Platform Suite. To migrate your ICAN 5.0 Projects, do the following:

Export the Project

1. Before you export your Projects, save your current ICAN 5.0 Projects to your Repository.
2. From the Project Explorer, right-click your Repository and select **Export** from the shortcut menu. The Export Manager appears.
3. From the **Select Projects from the list field** of the Export Manager, select one or more Projects that you want to export and move them 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, from the **Select Environments from the list** field, select the Environments that you want to export and move them 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 Sun Java Composite Application Platform Suite

7 Install Sun Java Composite Application Platform Suite, including all eWays, libraries, and other components used by your ICAN 5.0 Projects.

8 Start the Sun SeeBeyond Enterprise Designer.

Import the Project

9 From the Enterprise Designer's Project Explorer, right-click the Repository and select **Import Project** from the shortcut menu. The Import Manager appears.

10 Browse to and select your exported Project file.

11 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 Sun Java Composite Application Platform Suite 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.

12 Close the Import Manager after the Project is successfully imported.

Deploy the Project

13 You must create a new Deployment Profile for each of your imported Projects. When you export a Project, the Project's components are automatically *"checked in"* to Version Control to write-protect 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 **Check Out**.
- 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 **Check Out**.

14 If your imported Projects include File eWays, they must be reconfigured in your Environment prior to deploying the Project. To reconfigure your File eWays, do the following:

- A The Environment File External System properties can now accommodate both inbound and outbound eWays. If your previous Environment includes both inbound and outbound File External Systems, delete one of these (for example, the outbound File External System).

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

C The Directory property has been relocated from the Connectivity Map Properties to the Environment Properties. Set the inbound and outbound Directory values, and click **OK**.

15 Deploy your Projects.

Migrating 5.0 Projects that contain Japanese Character Encoding

The **e-Mail eWay version 5.1**, has changed the way in which Japanese Character encoding is configured. See **"Migrating ICAN 5.0 Projects that Use Japanese Encoding" on page 78** for directions on preparing your Project Collaborations to run on version 5.1.

Note: *Only projects developed on ICAN 5.0.2 and above can be imported and migrated successfully into the Java Integration Suite.*

Chapter 3

Configuring the e-Mail eWay Properties

This chapter describes how to create and configure the e-Mail eWay.

What's in This Chapter

3.1 Configuring the e-Mail eWay Properties

All eWays contain a set of parameters with properties that are unique to that eWay type. The e-Mail eWay properties are modified from these 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 e-Mail External System window.
- **Collaboration or Business Process**: Some e-Mail eWay properties can also be set from your Collaboration. Properties set from the Collaboration override the corresponding properties in the eWay's configuration file. Any properties that are not set from the Collaboration retain their configured default settings.

3.1.1 Selecting e-Mail as the External Application

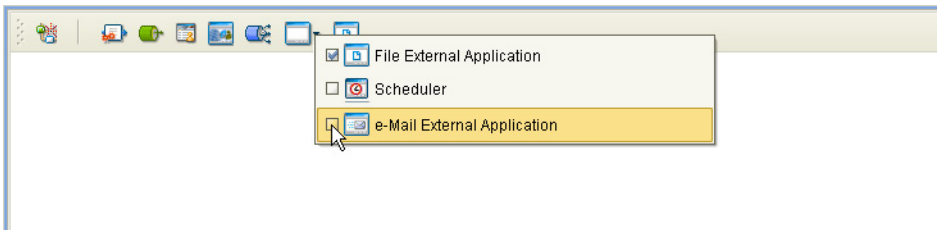
To create an e-Mail eWay you must first create an e-Mail External Application in your Connectivity Map. e-Mail eWays are located between an e-Mail External Application and a Service. Services are containers for Java Collaborations, Business Processes, eTL processes, and so forth.

To create the e-Mail External Application

1 From the Connectivity Map toolbar, click the External Applications icon.

2 Select the **e-Mail External Application** from the menu (see Figure 1). The selected e-Mail External Application icon appears on the Connectivity Map toolbar.

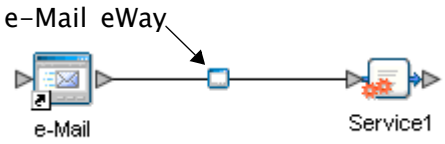
Figure 1 External Applications Selection Menu



3 Drag the new **e-Mail External Application** from the toolbar onto the Connectivity Map canvas. This represents an external e-Mail system.

From the Connectivity Map, you can associate (bind) the External Application with the Service to establish an eWay (see Figure 2).

Figure 2 eWay Location



When **e-Mail** is selected as the External Application, it automatically applies the default e-Mail eWay properties, provided by the OTD, to the eWay that connects it to the Service. These properties can then be modified for your specific system using the **Properties Editor**.

3.1.2 Modifying the e-Mail eWay Properties

A Project's eWay properties can be modified after the eWays have been established in the Connectivity Map and the Environment has been created.

Modifying the e-Mail eWay (Connectivity Map) Properties

The e-Mail eWay contains one Connectivity Map property (Polling Interval) for the inbound eWay. The outbound eWay contains no Connectivity Map properties.

1 From the Connectivity Map, double click the eWay icon, located in the link between the associated External Application and the Service.

2 The eWay **Properties Editor** opens with the e-Mail eWay Connectivity Map properties. Make any necessary modifications and click **OK** to save the settings.

Modifying the e-Mail eWay (Environment Explorer) Properties

1 From the Environment Explorer tree, right-click the e-Mail External System. Select **Properties** from the shortcut menu. The **Properties Editor** opens with the e-Mail eWay Environment properties.

2 Make any necessary modifications to the eWays Environment properties and click **OK** to save the settings.

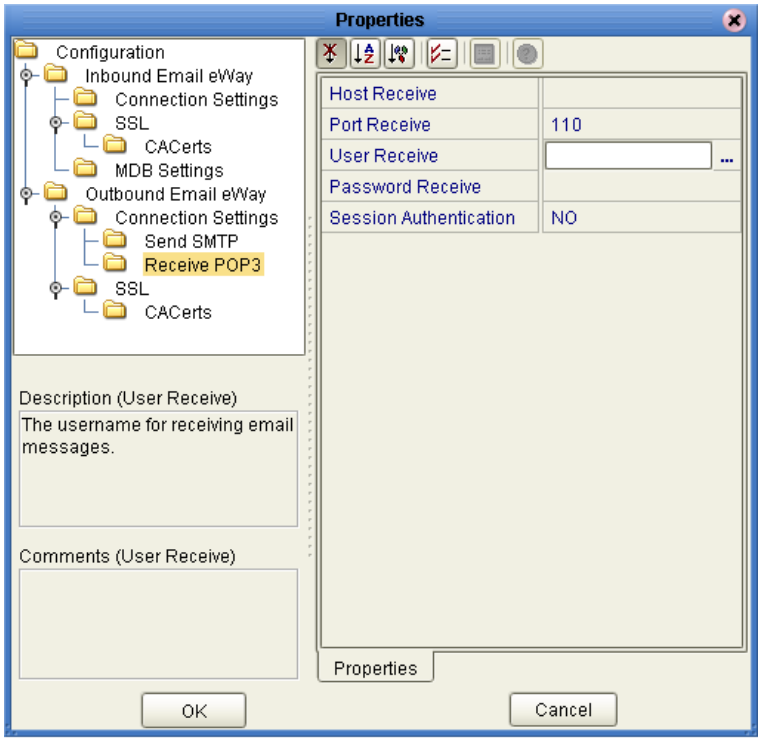
3.1.3 Using the Properties Editor

Modifications to the eWay properties are made using the e-Mail eWay Properties Editor.

Modifying the Default eWay Properties

1. An eWay has two different sets of parameters: those specific to that particular eWay (accessed from the **Connectivity Map**), and those that are common to all eWays of this type (accessed from the **Environment Explorer tree**). Open the e-Mail eWay's Environment properties by double-clicking the e-Mail External System in the Environment Explorer tree. The Properties Editor appears.
2. From the upper-right pane of the Properties Editor, select a subdirectory of the Environment Configuration directory. The parameters contained in that subdirectory are now displayed in the Properties pane of the Properties Editor. For example, clicking on the **Connector Settings** > Receive POP3 subdirectory displays that section's editable properties in the right pane, as shown in Figure 3.

Figure 3 Properties Editor -- e-Mail eWay Environment Properties



3. Click on any property field to make it editable. For example, click on the **User Receive** property to edit the user name. If a parameter's value is true/false or multiple choice, the field reveals a submenu of property options.

   Click on the ellipsis (. . .) in the properties field (displayed during modification of the value) to open a separate configuration dialog box. This is helpful for large values that cannot be fully displayed in the parameter's property field. Enter the property value in the dialog box and click **OK**. The value is now displayed in the parameter's property field.

4 A description of each parameter is displayed in the **Description** pane when that parameter is selected, providing an explanation of any required settings or options.

5 The **Comments** pane provides an area for recording notes and information regarding the currently selected parameter. This is saved for future referral.

6 After modifying the configuration properties, click **OK** to close the Properties Editor and save the changes.

3.2 e-Mail eWay Properties

The e-Mail eWay's Properties are organized as follows:

- **Inbound e-Mail eWay Connectivity Map Properties** on page 21
- **e-Mail eWay Environment Properties** on page 22

3.3 Inbound e-Mail eWay Connectivity Map Properties

The e-Mail eWay configuration parameters, accessed from the Connectivity Map, are organized into the following sections:

- **Polling Setting** on page 21

Note: *Some e-Mail eWay properties can also be set from your Collaboration. Properties set from the Collaboration override the corresponding properties in the eWay's configuration file. Any properties that are not set from the Collaboration retain their configured default settings.*

3.3.1 Polling Setting

The **Polling Setting** section of the e-Mail eWay Connectivity Map properties contains the top-level parameter displayed in Table 3.

Table 3 Connectivity Map - Polling Setting

Name	Description	Required Value
Polling Interval	Specifies the interval (in milliseconds) at which the e-mail source file is polled for new incoming e-mail messages.	A number indicating the polling interval in milliseconds. The configured default is **5000** (5 seconds).

3.4 e-Mail eWay Environment Properties

The e-Mail eWay configuration parameters, accessed from the e-Mail eWay External System in the Environment Explorer tree, are organized into the following sections:

Note: *Some e-Mail eWay properties can also be set from your Collaboration. Properties set from the Collaboration override the corresponding properties in the eWay's configuration file. Any properties that are not set from the Collaboration retain their configured default settings.*

3.4.1 Inbound Email eWay > Connection Settings

The **Inbound Email eWay > Connection Settings** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 4.

Table 4 Environment - Inbound Email eWay > Connection Settings

Name	Description	Required Value
Host Receive	Specifies the host name of the server used to receive messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the SessionAuth parameter is set to Yes (for POP3 login).	The host name of the server used to receive messages.
Port Receive	Specifies the port number used to connect when receiving e-mail messages. This is required for "receiving" eWay connections.This is also required for "sending" eWay connections when the SessionAuth parameter is set to Yes (for POP3 login).	The port number used to connect when receiving e-mail messages. This is a number between 1 and 65535. The configured default is 110.
User Receive	Specifies the user name used when receiving messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the SessionAuth parameter is set to Yes (for POP3 login).	The valid user login name used when receiving e-mail messages.
Password Receive	Specifies the password used when receiving messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the SessionAuth parameter is set to Yes (for POP3 login).	The user password used when receiving messages.

3.4.2 Inbound Email eWay > SSL

The Inbound Email eWay > SSL section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 5.

Table 5 Environment - Inbound Email eWay > SSL

Name	Description	Required Value
Receive SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the server. Please see the JSSE User's Guide for your Logical Host's platform. For more information on the supported SSL protocol options, see **"Using SSL" on page 80**	Select the appropriate SSL protocol. The options are: • **No SSL** • **TLS** • **TLSv1** • **SSLv3** • **SSLv2** • **SSL** The configured default is **No SSL**.
X509 Algorithm Name	Specifies the X509 algorithm name to use for the trust and key manager factories. See the JSSE User's Guide for your Logical Host's platform for more information.	An X509 algorithm name. The configured default is **SunX509**.

3.4.3 Inbound Email eWay > SSL > CACerts

The **Inbound Email eWay > SSL > CACerts** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 6.

Table 6 Environment - Inbound Email eWay > SSL > CACerts

Name	Description	Required Value
TrustStore type	Specifies the type of truststore used for CA certificate management when establishing SSL connections.	The trustStore type. The configured default is **JKS**.
TrustStore	Specifies a truststore used for CA certificate management to establish SSL connections. A truststore file is a key database file that contains the public keys for a target server.	The truststore used for CA certificate management.
TrustStore password	Specifies the password for accessing the truststore used for CA certificate management when establishing SSL connections.	The truststore password.

3.4.4 Inbound Email eWay > MDB Settings

The **Inbound Email eWay > MDB Settings** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 7.

Table 7 Environment - Inbound Email eWay > MDB Settings

Name	Description	Required Value
Max Pool Size	Specifies the maximum pool size. This controls the number of concurrent sessions.	An integer indicating the maximum pool size. The configured default is **10**.

3.4.5 Outbound Email eWay > Connection Settings > Send SMTP

The **Outbound Email eWay > Connection Settings > Send SMTP** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 8.

Table 8 Environment - Outbound Email eWay > Connection Settings > Send SMTP

Name	Description	Required Value
Host Send	Specifies the host name of the server used to send messages. This is required for the "sending" eWay connection.	The host name of the server used to send messages.
Port Send	Specifies the port number to connect to when sending messages. This is required for "sending" eWay connections.	An integer indicating the port number. The configured default is **25**.
User Send	Specifies the user name used when sending messages. This is required for "sending" eWay connections.	The login name used to access the sending host server.
Password Send	Specifies the password used when sending messages. This is required for "sending" eWay connections.	The user password used to access the sending host server.
Text encoding	Specifies the encoding used for body and header text. Available encoding options are: ▪ **ASCII**: for ASCII text. ▪ **iso-8859-1**: Latin 1 (Western Europe) text. ▪ **iso-2022-jp**: Japanese character text. ▪ **ISO2022CN**: Chinese character text. ▪ **ISO2022CN_GB**: Simplified Chinese character text. **ISO2022KR**: Korean character text.	Select one of the following: ▪ ASCII ▪ iso-8859-1 ▪ iso-2022-jp ▪ ISO2022CN ▪ ISO2022CN_GB ▪ ISO2022KR
Header encoding	Specifies the encoding used for the header. Available encoding options are: ▪ **B**: identical to the "BASE64" encoding defined by RFC 1341. ▪ **Q**: designed to allow text containing mostly ASCII characters to be deciphered by an ASCII terminal without decoding. Q encoding is similar to "Quoted-Printable" content- transfer-encoding defined in RFC 1341. "Q" encoding is recommended for use with most Latin character sets, while "B" encoding is recommended for all others.	Select one of the following: ▪ B ▪ Q

3.4.6 Outbound Email eWay > Connection Settings > Receive POP3

The **Outbound Email eWay > Connection Settings > Receive POP3** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 9.

Table 9 Environment - Outbound Email eWay > Connection Settings > Receive POP3

Name	Description	Required Value
Host Receive	Specifies the host name of the server used to receive messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the **SessionAuth** parameter is set to **Yes** (for POP3 login).	The host name of the server used to receive messages.
Port Receive	Specifies the port number to connect to when receiving messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connection when the **Session Authentication** parameter is set to **Yes** (for POP3 login).	An integer indicating the port number used to connect with the receiving host server. The configured default is **110**.
User Receive	Specifies the user name used when receiving messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the **Session Authentication** parameter is set to **Yes** (for POP3 login).	The login name used to access the receiving host server.
Password Receive	Specifies the password used when receiving messages. This is required for "receiving" eWay connections. This is also required for "sending" eWay connections when the **Session Authentication** parameter is set to **Yes** (for POP3 login).	The user password used to access the receiving host server.
Session Authentication	Determines whether a POP3 session authentication is performed before attempting an SMTP connection. This is required by some e-mail services. Set the value to **YES** only when necessary. **YES** requires that settings for **Host Receive**, **Port Receive**, **User Receive**, and **Password Receive** are entered for the **sending** eWay connection.	**YES** or **NO**. YES indicates that POP3 session authentication will be performed before attempting an SMTP connection. The configured default is **NO**.

3.4.7 Outbound Email eWay > SSL

The Outbound Email eWay > SSL section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 10.

Table 10 Environment - Outbound Email eWay > SSL

Name	Description	Required Value
Send SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the SMTP server. For more information on the supported SSL protocol options, see **"Using SSL" on page 80**	Select the appropriate SSL protocol. The options are: • **No SSL** • **TLS** • **TLSv1** • **SSLv3** • **SSLv2** • **SSL** The configured default is **No SSL**.
Receive SSL Protocol	Specifies the SSL protocol to use when establishing an SSL connection with the server. For more information on the supported SSL protocol options, see **"Using SSL" on page 80**	Select the appropriate SSL protocol. The options are: • **No SSL** • **TLS** • **TLSv1** • **SSLv3** • **SSLv2** • **SSL** The configured default is **No SSL**.
X509 Algorithm Name	Specifies the X509 algorithm name to use for the trust and key manager factories. See the JSSE User's Guide for your Logical Host's platform for more information.	An X509 algorithm name. The configured default is **SunX509**.

3.4.8 Outbound Email eWay > SSL > CACerts

The **Outbound Email eWay > SSL > CACerts** section of the e-Mail eWay Environment properties contains the top-level parameters displayed in Table 11.

Table 11 Environment - Outbound Email eWay > SSL > CACerts

Name	Description	Required Value
TrustStore type	Specifies the type of truststore used for CA certificate management when establishing SSL connections.	The trustStore type. The configured default is **JKS**.
TrustStore	Specifies a truststore used for CA certificate management to establish SSL connections.	The truststore used for CA certificate management.
TrustStore password	Specifies the password used to access the truststore used for CA certificate management when establishing SSL connections.	The truststore password.

Chapter 4

Using the e-Mail eWay with eInsight

This chapter describes how to use the e-Mail eWay with the Sun Java Composite Application Platform Suite's eInsight Business Process Manager and the Web Services interface.

Note: *You must have the* **eInsight.sar** *file installed to use the Web Services interface.*

What's in This Chapter

4.1 The eInsight Engine and Components

eGate components can be deployed as Activities in eInsight Business Processes. Using the Enterprise Designer with eInsight, you can add an Activity to a Business Process, then associate that Activity with an eGate component, such as an eWay. Then, when eInsight runs the Business Process, it automatically invokes that component using its Web Services interface. eGate components that can interface with eInsight in this way include:

- Object Type Definitions (OTDs)
- eWays
- Collaborations

See the *eInsight Business Process Manager User's Guide* for details.

4.2 The e-Mail eWay With eInsight

An eInsight Business Process Activity can be associated with the e-Mail eWay during the system design phase. To make this association, select the desired operators under

the eWay in the Enterprise Explorer and drag it onto the eInsight Business Process Designer canvas.

The e-Mail eWay has the following operators available:

- receive
- sendMessage
- hasMessage
- receiveMessage

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

4.3 e-Mail eWay Considerations

- Closely monitor the "sending" e-mail account for any errors in delivery. All error messages for errors in delivery that occur after the initial mail server will be sent only to the sending mail account.
- In the Collaboration or Business Process, when sending attachments, one of the following combinations must be mapped (set).
  - **content** and **contentType**
  - **name** and **contentType**
  - **content**, **contentType**, and **name**

  The content is a byte array (byte[]), and contentType and name Strings (java.lang.String).
- The E-mail eWay does not currently support sending or receiving messages containing multi-level nested MIME parts.

4.4 The e-Mail eWay eInsight Sample Project

This following pages provide directions for creating a simple Project that demonstrates how eInsight Business Processes are used with the e-Mail eWay. The same Project, **prjEmail_BPEL,** can be uploaded and imported in a near-complete state.

For more information on creating Java Integration Suite Projects see the *Sun SeeBeyond eInsight™ Business Process Manager User's Guide* and the *Sun SeeBeyond eGate™ Integrator User's Guide*.

4.5 Importing a Sample Project

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 Java Composite Application Platform Suite Installer's Documentation tab. The **EmaileWay_Sample.zip** file contains the various sample Project zip files. 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 (for this sample, **prjEmail_BPEL.zip**) and click **Import**. After the sample Project is successfully imported, click **Close**.
5. Before an imported sample Project can be run you must do the following:
   - Create an Environment (see **"Creating an Environment" on page 46**)
   - Configure the eWays for your specific system (see **"Configuring the eWays" on page 46**)
   - Create a Deployment Profile (see **"Creating and Activating the Deployment Profile" on page 49**)
   - Create and start a domain (see **"Creating and Starting the Domain" on page 50**)
   - Build and deploy the Project (see **"Building and Deploying the Project" on page 51**)
   - The following pages provide step by step directions for manually creating the prjEmail_BPEL Project.

4.6 The prjEmail_BPEL Project Overview

The e-Mail eWay Project, **prjEmail_BPEL**, demonstrates the following:

Send Message

- The inbound File eWay subscribes to an external directory. When a message is present, the File eWay picks up the message (an XML file containing the ToAddress, CCAddress, FromAddress, subject, and message text) and publishes the message to the bpSend Business Process.
- The bpSend Business Process formats the ToAddress, CCAddress, FromAddress, subject, and message, and publishes the message to the outbound e-Mail eWay.
- The outbound e-Mail eWay publishes the e-mail message to a SMTP server.

Receive Message

- The inbound e-Mail eWay subscribes to an external e-mail account, polling the account every 5 seconds. When e-mail is present, the e-Mail eWay publishes the message to the bpReceive Business Process.
- The bpReceive Business Process parses the e-mail message, and writes only the message text to a text file.
- The outbound File eWay writes the message text to an external directory.

Read Message

- The inbound File eWay, acting as a trigger, subscribes to an external directory. When a message is present, the File triggers the bpRead Business Process.
- The e-Mail eWay, triggered by the bpRead Business Process, subscribes to the email server. If a message is present, the e-Mail eWay publishes the message to the bpRead Business Process.
- The bpRead Business Process parses the ToAddress, CCAddress, FromAddress, subject, and message.
- The outbound File eWay writes this information to an external directory.

Receive Message checks for e-mail continuously, while Read Message only checks for e-mail when the Business Process is triggered by a file.

4.7 Creating the prjEmail_BPEL Project

The prjEmail_BPEL Project demonstrates the same operations as the prjEmail_JCD sample Project. The difference between the two Projects is that the prjEmail_BPEL uses Business Processes to perform the business logic, where as the prjEmail_JCD uses Java Collaborations to perform the business logic.

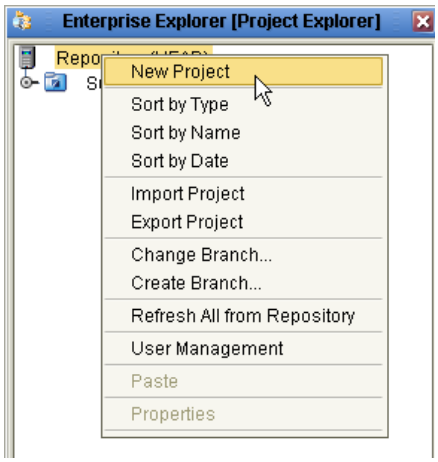
The following pages provide step by step directions for creating the prjEmail_BPEL Project.

4.7.1 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** (see **Figure 4 on page 35**). A new Project (**Project1**) appears on the Project Explorer tree.

Figure 4 Enterprise Explorer - New Project



3 Rename the Project (for this sample, **prjEmail_BPEL**).

4.7.2 Creating a Connectivity Maps

The Connectivity Map provides a canvas for assembling and configuring a Project's components.

1 From the Project Explorer tree, right-click the new **prjEmail_BPEL** 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 **cmSend**.

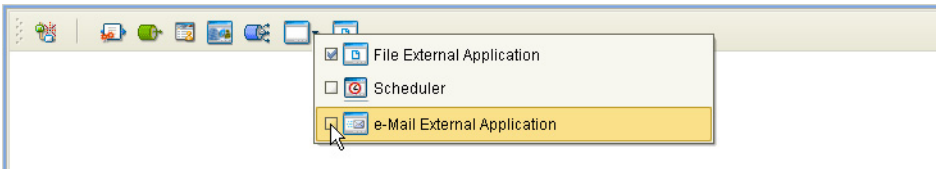
3 Create two additional Connectivity Maps named **cmRead** and **cmReceive**.

The icons in the toolbar represent the available components used to populate the Connectivity Map canvas.

Selecting the External Applications

In the Connectivity Maps, eWays are associated with External Systems. For example, to establish a connection to an external e-Mail application, you must first select e-Mail as an External Application to use in your Connectivity Map (see Figure 5).

Figure 5 Connectivity Map - External Applications



1 Click the **External Application** icon on the Connectivity Map toolbar,

2 Select the External Applications you require to create your Project (for this sample, **File** and **e-Mail**). Icons representing the selected External Applications are added to the Connectivity Map toolbar.

Populating the Connectivity Maps

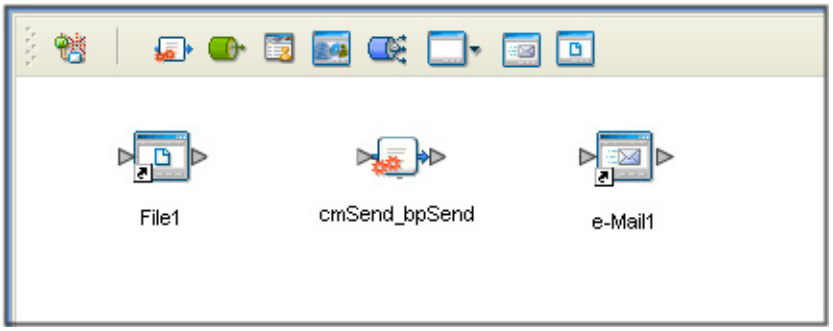
For the **cmSend** Connectivity Map, add the Project components by dragging the icons from the toolbar to the canvas.

1 For this sample, drag the following components onto the Connectivity Map canvas as displayed in Figure 6.

- e-Mail External System
- Service (A service is a container for Java Collaborations, Business Processes, eTL processes, and so forth)
- File External System

2 Rename the cmSend_Service1 to **cmSend_bpSend** (see Figure 6):

Figure 6 cmSend Connectivity Map

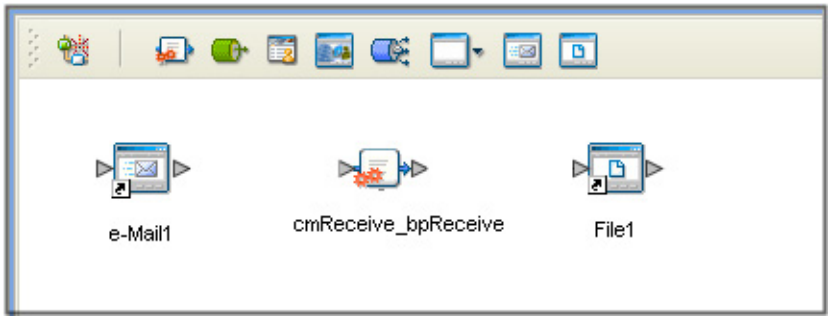


3 Populate the **cmReceive** Connectivity Map with the following components, as displayed in Figure 7:

- File External System
- Service (rename to **cmReceive_bpReceive**)
- e-Mail External System

To do this, drag and drop the components from the Connectivity Map toolbar to the Connectivity Map canvas.

Figure 7 cmReceive Connectivity Map

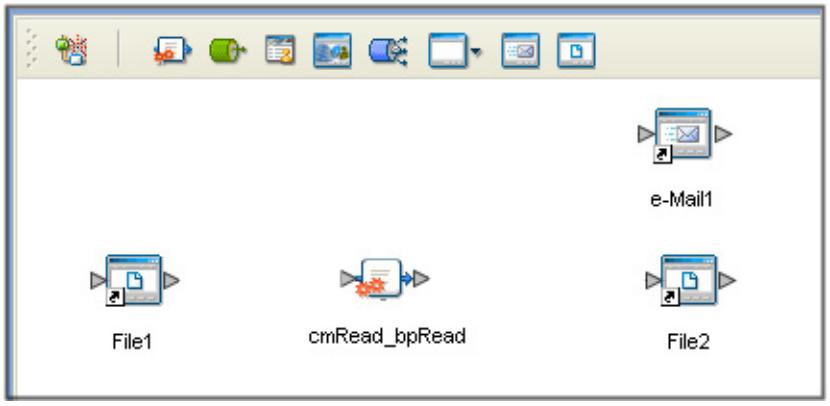


4 Populate the **cmRead** Connectivity Map with the following components, as displayed in **Figure 8 on page 37**:

- File External System (2)

- Service (rename to **bpRead_bpRead**)
- e-Mail External System

Figure 8 cmRead Connectivity Map



5 Save your current changes to the Repository.

4.7.3 Creating an OTD Using the Object Type Definition Wizard

The prjEmail_BPEL Project contains two Business Processes that utilize a user-created Object Type Definition (OTD). This OTD is created from a Document Type Definition (DTD) using the Object Type Definition Wizard. A DTD file, **eMail.dtd**, is provided with the sample Project file. To create the **eMail_SendEmail** OTD, do the following:

1 From the Project Explorer, right-click the **prjEmail_BPEL** Project and select New > Object Type Definition. The Object Type Definition Wizard appears.

2 For step 1 of the wizard, from the Select Wizard Type window, select **DTD**, and click **Next**.

3 For step 2 of the wizard, from the Browse DTD Files window, locate and select the **eMail.dtd** file that you downloaded to your computer with the e-Mail eWay's sample Projects. Click **Select** and click **Next**.

4 For step 3 of the wizard, from the Select Document Elements field, make sure that **eMail_SendEmail** is selected and click **Next**.

5 For step 4 of the wizard, keep all of the default OTD options, and click **Finish**. The OTD Editor appears with the new **eMail_SendEmail** OTD, and the OTD is added to the prjEmail_BPEL Project.

4.7.4 Creating a Business Process

The next step is to create the Project's three Business Processes:

- bpSend
- bpReceive
- bpRead

To create the Business Processes, do the following:

Creating the bpSend Business Process

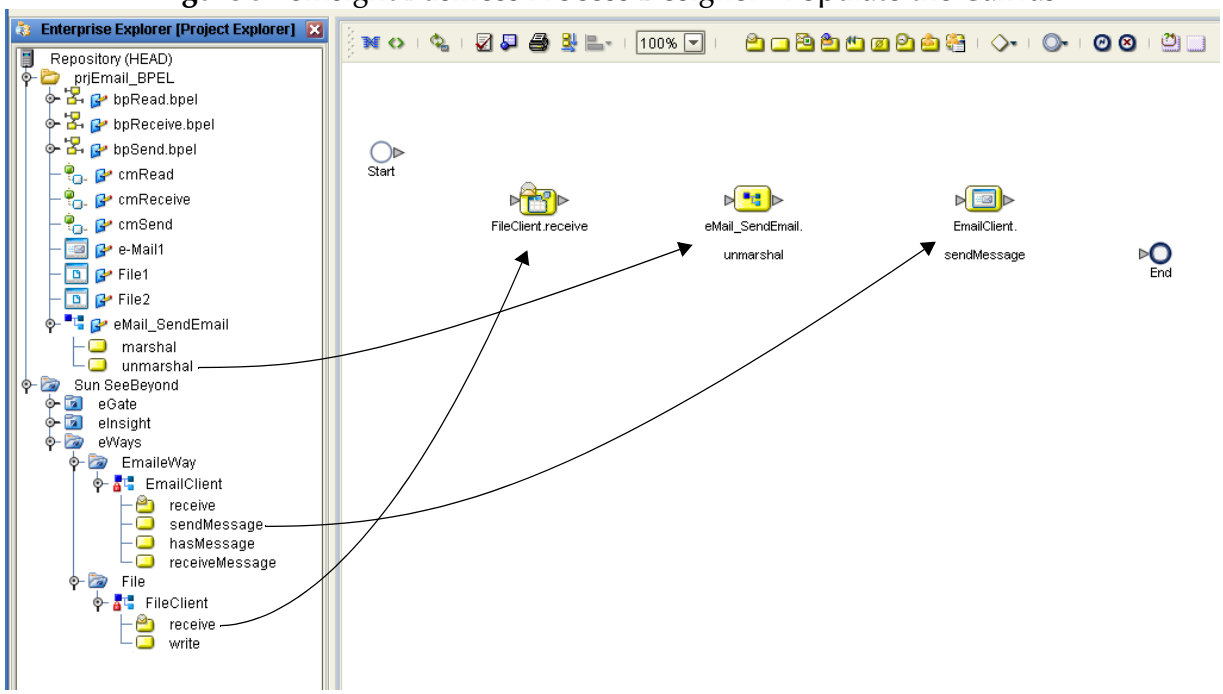
1 Right-click the **prjEmail_BPEL** Project in the Enterprise Designer's Project Explorer, and select **New** > **Business Process** from the shortcut menu. The eInsight Business Process Designer appears and **BusinessProcess1** is added to the Project Explorer tree. Rename **BusinessProcess1** to **bpSend**.

2 From the Project Explorer tree, expand the **Sun SeeBeyond > eWays > EmaileWay > EmailClient, and File > FileClient** nodes.

3 Populate the eInsight Business Process Designer's modeling canvas with the following activities from the Project Explorer tree as displayed in Figure 9:

- **receive**, located under **Sun SeeBeyond > eWays > File > FileClient**
- **unmarshal**, located under **prjEmail_BPEL > eMail_SendEmail** (OTD)
- **sendMessage**, located under **Sun SeeBeyond > eWays > EmaileWay > EmailClient**

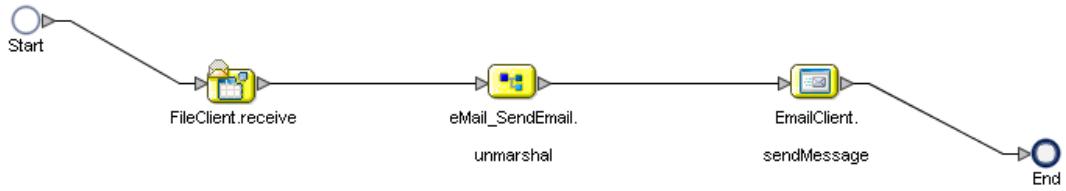
Figure 9 eInsight Business Process Designer - Populate the Canvas



4 Link the modeling elements by clicking on the element connector and dragging the cursor to the next element connector, making the following links as displayed in Figure 10 on page 39.

- **Start** to **FileClient.receive**
- **FileClient.receive** to **eMail_SendEmail.unmarshal**
- **eMail_SendEmail.unmarshal** to **EmailClient.sendMessage**
- **EmailClient.sendMessage** to **End**

Figure 10 Business Process Designer - Link the Modeling Elements



Configuring the bpSend Modeling Elements

Business Rules, created between the Business Process Activities, allow you to configure the relationships between the input and output Attributes of the Activities using the Business Process Designer's Business Rule Designer.

Adding Business Rules

1 Right-click the link between the **FileClient.receive** and **eMail_SendEmail.unmarshal** Activities and select **Add Business Rule** from the shortcut menu (see Figure 11).

Figure 11 eInsight Business Process Designer - Adding Business Rules

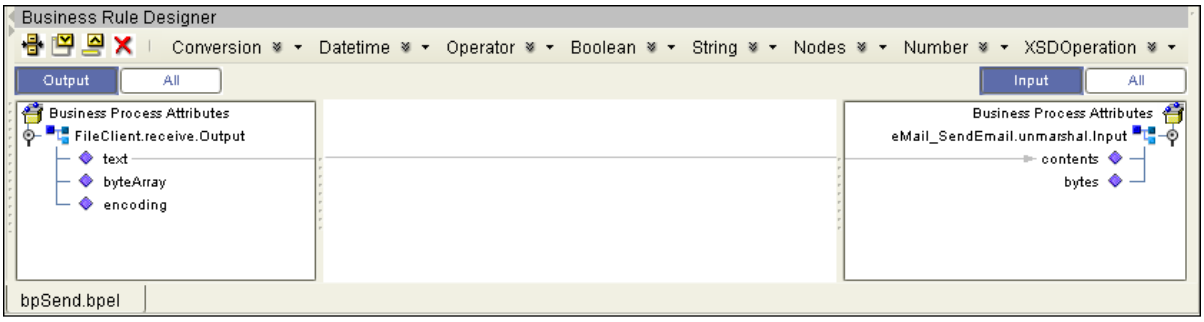
Using the Business Rule Designer

2 From the eInsight Business Process Designer toolbar, click the **Display Business Rule Designer** button. The Business Rule Designer appears at the bottom of the eInsight Business Process Designer.

3 Click on the Business Rule icon in the link between **FileClient.receive** and **eMail_SendEmail.unmarshal** to display the Business Rule's Input and Output Attributes in the Business Rule Designer. These Attributes can now be modified.

4 Map **text**, under **FileClient.receive.Output** in the Output pane of the Business Rule Designer, to **contents** under **eMail_SendEmail.unmarshal.Input** in the Input pane of the Business Rule Designer. To do this, click on **text** in the Output pane and drag the cursor to **contents** in the Input pane. A link now connects the two nodes in the Business Rule Designer (see **Figure 12 on page 40**).

Figure 12 Business Rule Designer - bpSend Business Process

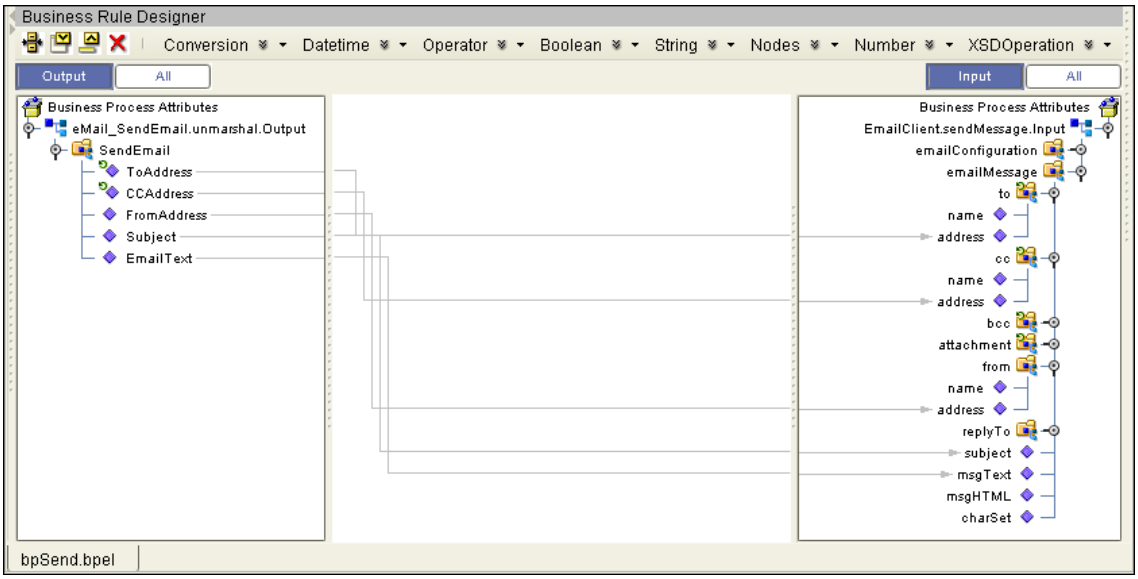


5 Add another Business Rule in the link between **eMail_SendEmail.unmarshal** and **EmailClient.sendMessage**. Click on the new Business Rule icon between **eMail_SendEmail.unmarshal** and **EmailClient.sendMessage** to display the Business Rule's Input and Output Attributes.

6 From the Business Rule Designer, map the following elements from the Output pane (under eMail_SendEmail.unmarshal.Output), to the Input pane (under **EmailClient.sendMessage.Input),** as displayed in Figure 13:

- **ToAddress** to **address** under **emailMessage > to**
- **CCAddress** to **address** under **emailMessage > cc**
- **FromAddress** to **address** under **emailMessage > from**
- **Subject** to **subject** under **emailMessage > replyTo**
- **EmailText** to **msgText** under **emailMessage > replyTo**

Figure 13 Business Rule Designer - bpSend Business Process



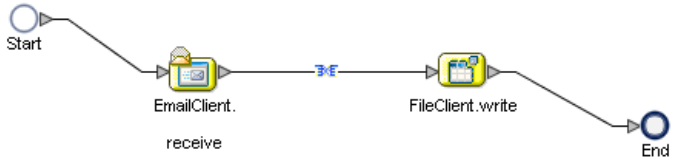
7 From the Business Process Designer toolbar, click the **Synchronize Graphical Model and Business Process** icon to synchronize the graphical interface to the Business Process code.

8 Save your changes to the Repository.

Creating the bpReceive Business Process

1. Right-click the 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 **bpReceive**.
2. Populate the eInsight Business Process Designer's modeling canvas with the following activities from the Project Explorer tree as displayed in Figure 14:
   - **receive**, under **Sun SeeBeyond > eWays > EmaileWay > EmailClient**
   - **write**, under **Sun SeeBeyond > eWays > File > FileClient**
3. Link the modeling elements by clicking on the element connector and dragging the cursor to the next element connector, making the following links as displayed in Figure 14.
   - **Start** to **EmailClient.receive**
   - **EmailClient.receive** to **FileClient.write**
   - **FileClient.write** to **End**

Figure 14 Business Process Designer - bpReceive



Configuring the bpReceive Modeling Elements

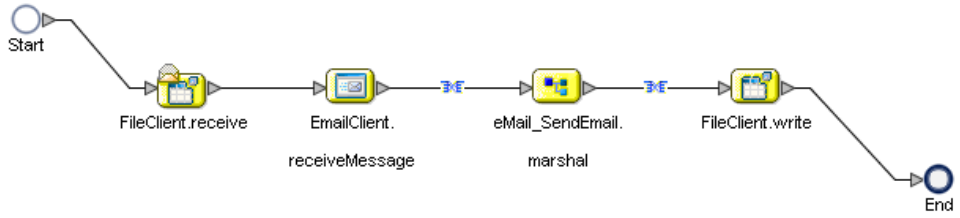
To create the **bpReceive** Business Rules do the following:

1. Right-click the link between the **EmailClient.receive** and **FileClient.write** Activities and select **Add Business Rule** from the shortcut menu.
2. From the eInsight Business Process Designer toolbar, click the **Display Business Rule Designer** button. The Business Rule Designer appears at the bottom of the eInsight Business Process Designer.
3. Click on the Business Rule icon in the link between **EmailClient.receive** and **FileClient.write** to display the Business Rule's Input and Output Attributes in the Business Rule Designer. These Attributes can now be modified.
4. Map **msgText**, under **EmailClient.receive.Output** > **emailMessage** in the Output pane of the Business Rule Designer, to **text** under **FileClient.write.Input** in the Input pane of the Business Rule Designer.
5. From the Business Process Designer toolbar, click the **Synchronize Graphical Model and Business Process** icon to synchronize the graphical interface to the Business Process code.
6. Save your changes to the Repository.

Creating the bpRead Business Process

1. Right-click the Project in the Enterprise Designer's Project Explorer, and select **New > Business Process** from the shortcut menu. Rename **BusinessProcess1** to **bpRead**.
2. Populate the eInsight Business Process Designer's modeling canvas with the following activities from the Project Explorer tree as displayed in Figure 15:
   - **receive**, under **Sun SeeBeyond > eWays > File > FileClient**
   - **receiveMessage**, under **Sun SeeBeyond > eWays > EmaileWay > EmailClient**
   - **marshal**, located under **prjEmail_BPEL > eMail_SendEmail** (OTD)
   - **write**, under **Sun SeeBeyond > eWays > File > FileClient**
3. Link the modeling elements by clicking on the element connector and dragging the cursor to the next element connector, making the following links as displayed in Figure 15.
   - **Start** to **FileClient.receive**
   - **FileClient.receive** to **EmailClient.receiveMessage**
   - **EmailClient.receiveMessage** to **eMail_SendEmail.marshal**
   - eMail_SendEmail.marshal to FileClient.write
   - **FileClient.write** to **End**

Figure 15 Business Process Designer - bpRead



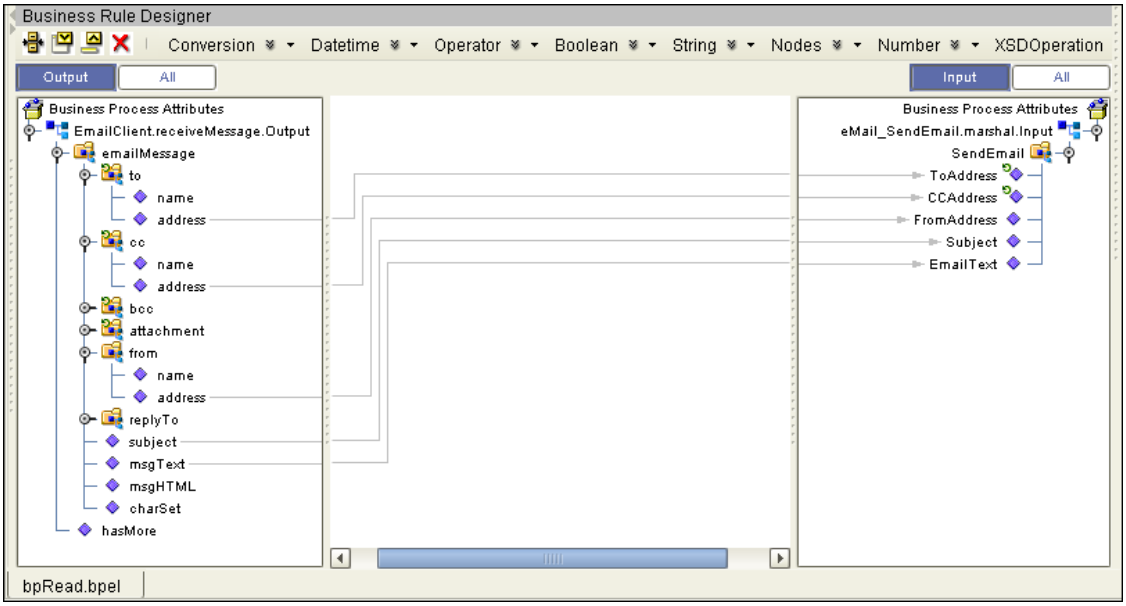
Configuring the bpRead Modeling Elements

To create the **bpRead** Business Rules do the following:

1. Right-click the link between the **EmailClient.receiveMessage** and **FileClient.write** Activities and select **Add Business Rule** from the shortcut menu.
2. From the eInsight Business Process Designer toolbar, click the **Display Business Rule Designer** button. The Business Rule Designer appears at the bottom of the eInsight Business Process Designer.
3. From the Business Rule Designer, map the following elements from the Output pane (under **EmailClient.receiveMessage.Output**), to the Input pane (under **eMail_SendEmail.marshal.Input > SendEmail),** as displayed in Figure 16:
   - **address under emailMessage > to ToAddress**
   - **address under emailMessage > cc to CCAddress** to **address**
   - **address under emailMessage > from to FromAddress**

- **subject under emailMessage > replyTo to Subject**
- **msgText under emailMessage > replyTo to EmailText**

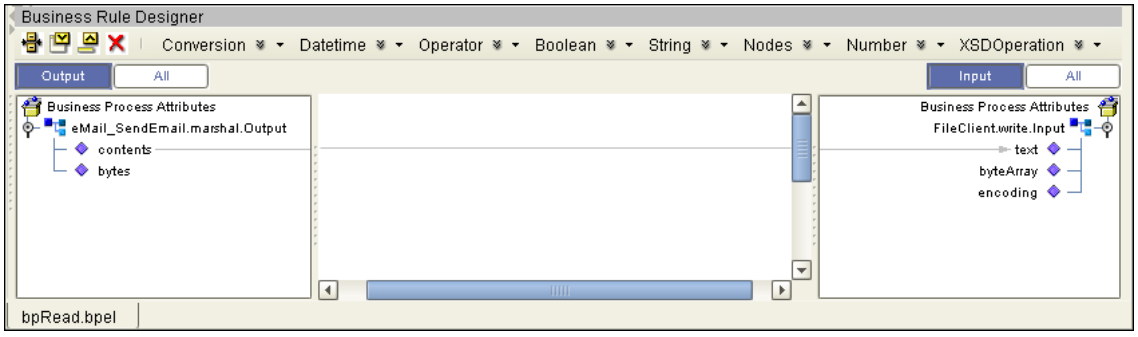
Figure 16 Business Rule Designer - bpRead Business Process



4 Add another Business Rule in the link between **eMail_SendEmail.marshal** and **FileClient.write**. Click on the new Business Rule icon between **eMail_SendEmail.marshal** and **FileClient.write** to display the Business Rule's Input and Output Attributes.

5 From the Business Rule Designer, map **content** in the Output pane under **eMail_SendEmail.marshal.Output**, to text in the Input pane under **FileClient.write.Input),** as displayed in Figure 17.

Figure 17 Business Rule Designer - bpRead Business Process



6 From the Business Process Designer toolbar, click the **Synchronize Graphical Model and Business Process** icon to synchronize the graphical interface to the Business Process code.

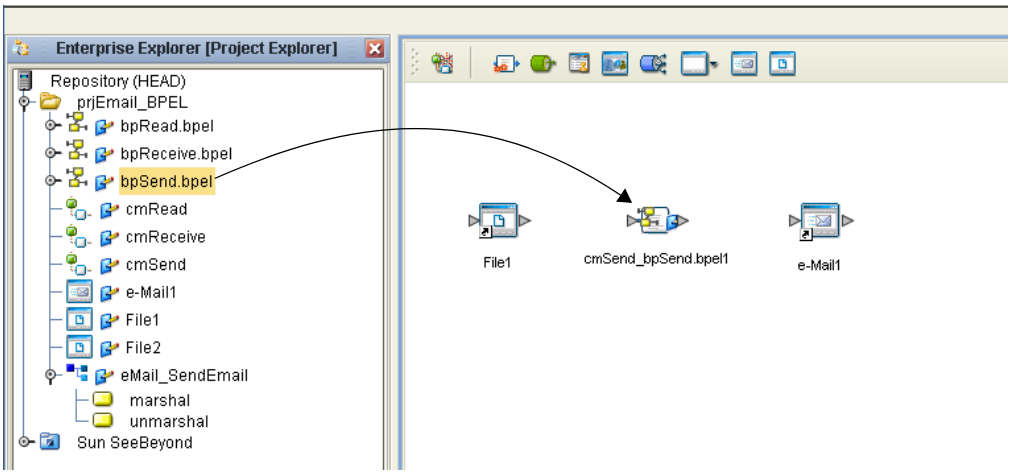
7 Save your changes to the Repository.

4.7.5 Binding the eWay Components

After the Business Process have been created, the components are associated and bindings are created in the Connectivity Maps.

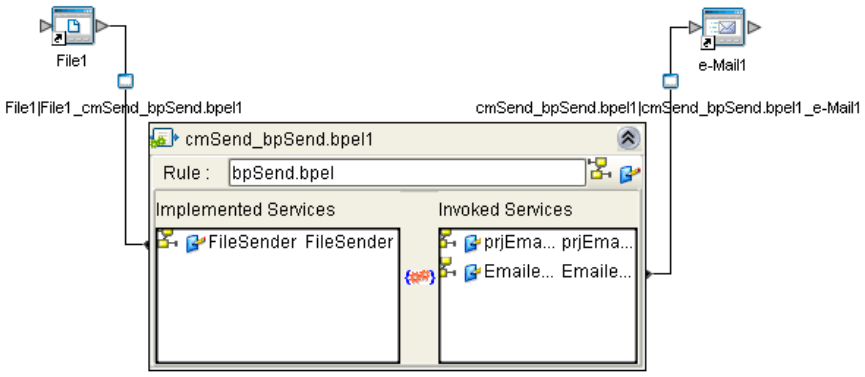
1. From the Project Explorer, double-click **cmSend** to display the **cmSend** Connectivity Map.
2. Drag and drop the **bpSend** Business Process from the Project Explorer to the **cmSend_bpSend** Service. If the Business Process was successfully associated, the Service icon changes to a Business Process (see Figure 18).

Figure 18 Connectivity Map - Binding the Components



3. From the Connectivity Map canvas, double-click **cmSend_bpSend**. The **cmSend_bpSend** binding dialog box appears using the **bpSend** Rule.
4. From the **cmSend_bpSend** binding dialog box, map **FileClient Input** (under Implemented Services) to the output node of the inbound **File1** External Application. To do this, click on **FileClient Input** under Implemented Services in the cmSend_bpSend binding box, and drag your cursor to the output node of the **File1** External Application. A link now joins the two components.
5. From the **cmSend_bpSend** binding dialog box, map **EmailClient** (under Invoked Services) to the input node of the **e-Mail1** External Application (see **Figure 19 on page 45**).

Figure 19 Connectivity Map - Associating (Binding) the Project's Components



6 Minimize the **cmSend_bpSend** binding dialog box.

7 From the Project Explorer, double-click **cmReceive** to display the **cmReceive** Connectivity Map.

8 Drag and drop the **bpReceive** Business Process from the Project Explorer to the **cmReceive_bpReceive** Service.

9 From the Connectivity Map, double-click the **cmReceive_bpReceive** Service. The **cmReceive_bpReceive** binding dialog box appears with the **bpReceive** Rule.

10 From the **cmReceive_bpReceive** binding dialog box, map **EmailClient input** (under Implemented Services) to output node of the **e-Mail1** External Application.

11 From the **cmReceive_bpReceive** binding dialog box, map **FileClient_1, FileClient** (under Invoked Services) to the input node of the outbound **File1** External Application.

12 Minimize the **cmReceive_bpReceive** binding dialog box.

13 From the Project Explorer, double-click **cmRead** to display the **cmRead** Connectivity Map.

14 Drag and drop the **bpRead** Business Process from the Project Explorer to the **cmRead_bpRead** Service.

15 From the Connectivity Map, double-click the **cmRead_bpRead** Service. The **cmRead_bpRead** binding dialog box appears with the **bpRead** Rule.

16 From the **cmRead_bpRead** binding dialog box, map **FileClient input** (under Implemented Services) to the output node of the **File1** External Application.

17 From the **cmRead_bpRead** binding dialog box, map **EmailClient_1 EmailClient** (under Invoked Services) to the input node of the **e-Mail1** External Application.

18 From the **cmRead_bpRead** binding dialog box, map **FileClient_1 FileClient** (under Invoked Services) to the input node of the outbound **File2** External Application.

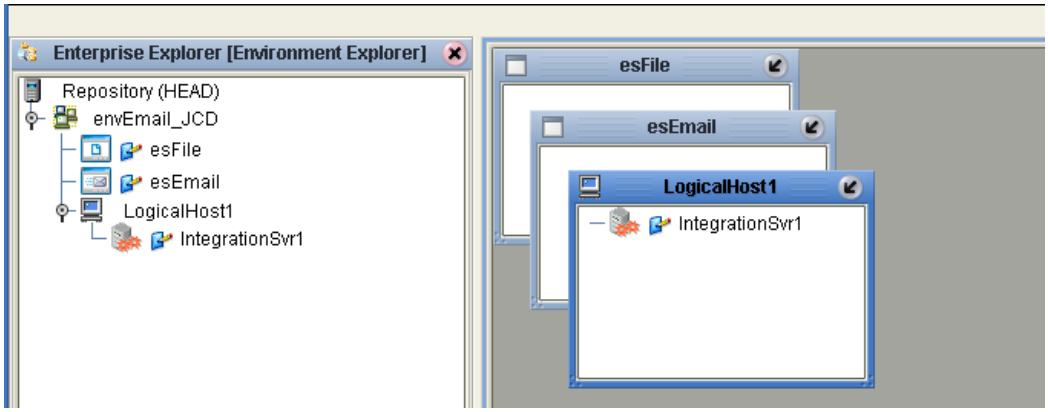
19 Minimize the **cmRead_bpRead** binding dialog box, and save all your current changes to the Repository.

4.7.6 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 Explorer and 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 **envEmail_BPEL**.
4. Right-click **envEmail_BPEL** and select **New > e-Mail External System**. Name the External System **esEmail** and click **OK**. **esEmail** is added to the Environment Editor.
5. Right-click **envEmail_BPEL** and select **New > File External System**. Name the External System **esFile** and click **OK**. **esFile** is added to the Environment Editor.
6. Right-click **envEmail_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 20.

Figure 20 Environment Editor - envEmail_BPEL

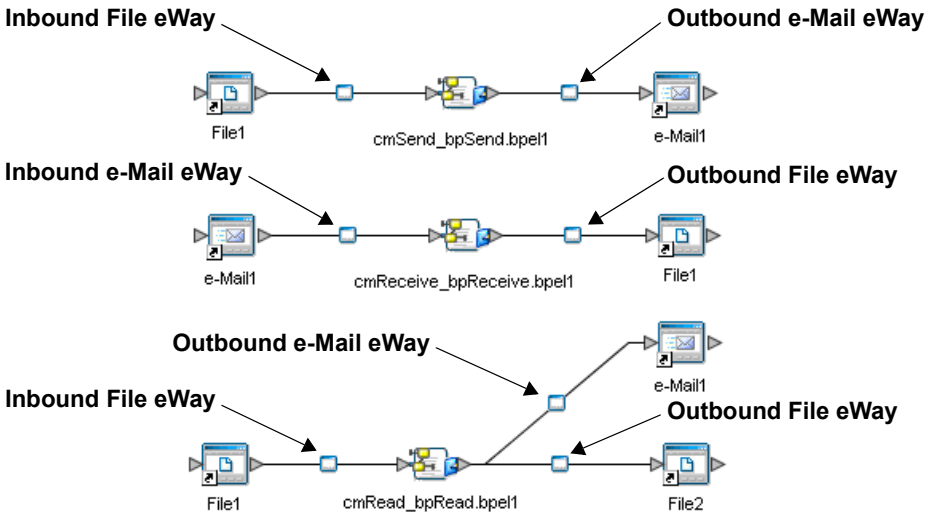


9. Save your current changes to the Repository.

4.7.7 Configuring the eWays

The prjEmail_BPEL Project uses four eWays, each represented in the Connectivity Maps as a node between an External Application and a Service. eWays facilitate communication and movement of data between the external applications and the eGate system.

Figure 21 eWays



The eWay properties are set from both the Project Explorer's Connectivity Maps and the Environment Explorer tree.

Configuring the File eWay Properties

1. From the cmSend Connectivity Map, double-click the inbound **File1** eWay (see Figure 21). 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 bpSend - Inbound File eWay Settings

Inbound File eWay Connection Parameters	
Input file name	email.xml

3. In the same way, modify the properties of the cmRead Connectivity Map's inbound **File1** eWay, entering **trigger.txt** as the Input file name property value.
4. From the cmReceive Connectivity Map, modify the outbound **File1** eWay properties for your system, including the settings in Table 13.

Table 13 Outbound File eWay Settings

Outbound File eWay Connection Parameters	
Output file name	output%d.dat

5. In the same way, modify the properties of the cmRead Connectivity Maps outbound **File2** eWay.

6 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.

7 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:/temp*
Outbound File eWay > Parameter Settings Set as directed, otherwise use the default settings	
Directory	*Select a directory, for example C:/temp*

Configuring the e-Mail eWay Properties

1 The default e-Mail eWay **Connectivity Map** properties are sufficient for this sample.

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

3 Modify the e-Mail eWay Environment properties for your system, including the settings in Table 15.

Table 15 e-Mail eWay Environment Explorer Properties

e-Mail eWay Environment Properties	
Inbound Email eWay > Connection Settings Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Port Receive	*TCP/IP Port Number for receiving e-mail*
User Receive	*User name for receiving e-mail*
Password Receive	*User password for receiving e-mail*
Outbound Email eWay > Connection Settings > Send SMTP Set as directed, otherwise use the default settings	
Host Send	*Host name of the sending SMTP server*
Port Send	*TCP/IP Port Number used for sending e-mail*
User Send	*User name for sending e-mail*
Password Send	*The password for sending email messages.*
Outbound Email eWay > Connection Settings > Receive POP3 Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Port Receive	*TCP/IP Port Number used for receiving e-mail*

e-Mail eWay Environment Properties	
User Receive	*User name for receiving e-mail*
Password Receive	*The password for receiving email messages.*
Session Authentication	*NO*

4.7.8 Configuring the Integration Server

You must set your Sun 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. Enter **STC** as the **Specific Value** and as the **Confirm Password**, and click **OK**.
4. 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*.

4.7.9 Creating and Activating the Deployment Profile

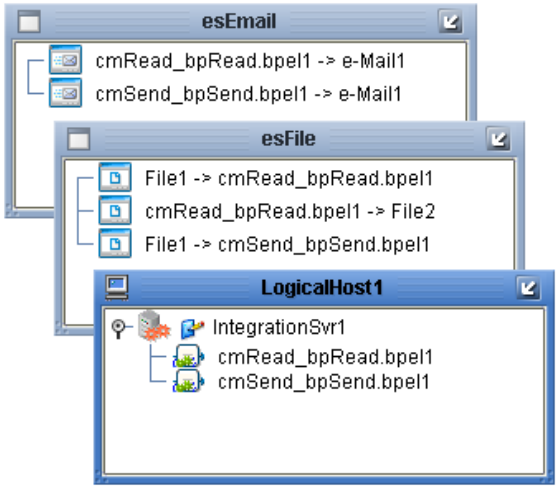
Deployment Profiles are specific instances of a Project in a particular Environment. A Deployment Profile contains information regarding the assignment of services and message destinations to integration and message servers (JMS IQ Managers). It also contains version information for all versionable objects in the Project. Deployment Profiles are created using the Deployment Editor.

The prjEmail_BPEL Project performs multiple operations and requires two different Deployment Profiles; **dp_bpRead**, and **dp_bpReceive**. This allows part of the Project to be undeployed.

Create the dp_bpRead Deployment Project

1. From the Project Explorer, right-click the **prjEmail_BPEL** Project and select **New** > **Deployment Profile** from the shortcut menu. The **Create Deployment Profile for prjEmail_BPEL** dialog box appears.
2. Enter **dp_bpRead** as the name for the Deployment Profile. Select **envEmail_BPEL** as the Environment, and make sure that only the **cmSend** and **cmRead** Connectivity Maps are selected (checked). Click **OK**. The Deployment Editor appears.
3. From the Deployment Editor, click **Automap**. This automatically maps all of your Project components to the correct External Systems and Integration Server. The Automap Results dialog box appears. Click **Close**. The **dp_bpRead** Deployment Profile now contains the components displayed in Figure 22.

Figure 22 dp_bpRead Deployment Profile



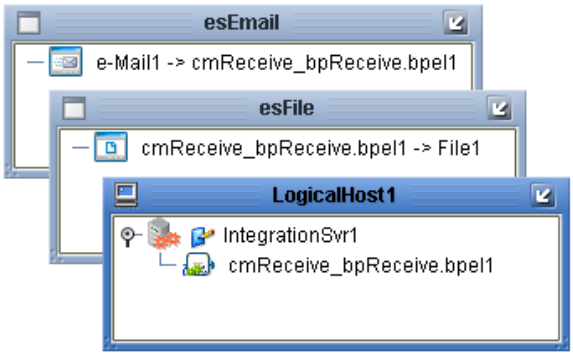
Create the dp_bpReceive Deployment Project

1 From the Project Explorer, right-click the **prjEmail_BPEL** Project and select **New** > **Deployment Profile** from the shortcut menu. The **Create Deployment Profile for prjEmail_BPEL** dialog box appears.

2 Enter **dp_bpReceive** as the name for the Deployment Profile. Select **envEmail_BPEL** as the Environment, and make sure that only the **cmMQReceive** Connectivity Map is selected (checked). Click **OK**.

3 From the Deployment Editor, click **Automap**. Review and close the Automap Results dialog box. The **dp_bpRead** Deployment Profile now contains the components displayed in Figure 23.

Figure 23 dp_bpReceive Deployment Profile



4 Save your current changes to the Repository.

4.7.10 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 Integration 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.

4.7.11 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.*

4.7.12 Running the Sample

The prjEmail_BPEL Project Demonstrates three different operations (see **"The prjEmail_BPEL Project Overview" on page 33**).

To run the **Send Message** and **Read Message** operations, the **dp_bpReceive** Deployment Profile must be undeployed from the Sun SeeBeyond Enterprise Manager.

To run the **Send Message** and **Receive Message** operations, the **dp_bpRead** and **dp_bpReceive** Deployment Profiles are deployed, and the **Receive Message** operation supersedes the **Read Message** operation.

For information on deploying and undeploying your deployments using the Sun SeeBeyond Enterprise Manager, see the *Sun SeeBeyond eGate Integrator System Administration Guide*.

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.

Chapter 5

Using the e-Mail eWay with Java Collaborations

This chapter provides an introduction to the e-Mail eWay components and information on how these components

are created and implemented in an eGate Project. It is assumed that the reader understands the basics of creating a Project using the Sun Java Composite Application Platform Suite. For more information on creating an eGate Project see the *Sun SeeBeyond eGate™ Tutorial* and the *Sun SeeBeyond eGate™ Integrator User's Guide*.

What's in This Chapter

5.1 e-Mail eWay Components

This chapter presents a sample e-Mail eWay Project created using the same procedures as the sample end-to-end Project provided in the *Sun SeeBeyond eGate Tutorial*. Components that are unique to the e-Mail eWay include the following:

e-Mail eWay Properties File

The properties file for the e-Mail eWay contains the parameters that are used to connect with a specific external system. These parameters are set using the Properties Editor. For more information about the e-Mail eWay properties file and the Properties Editor see **"Configuring the e-Mail eWay Properties" on page 18**.

EmailMessage OTD

The EmailMessage OTD is provided with the eWay and contains methods and attributes that are used to create the Business Rules that invoke the e-Mail program.

5.2 e-Mail eWay Considerations

- Closely monitor the "sending" e-mail account for any errors in delivery. All error messages for errors in delivery that occur after the initial mail server will be sent only to the sending mail account.
- After a successful call to **EmailMessage.receiveMessage()** and before a successful call to **send()**, the received e-mail message is contained ONLY in the EmailMessage instance in the Collaboration Rule. Failure to process and/or store it properly will result in message loss.
- In the Collaboration or Business Process, when sending attachments, one of the following combinations must be mapped (set).
  - **content** and **contentType**
  - **name** and **contentType**
  - **content**, **contentType**, and **name**

  The content is a byte array (byte[]), and contentType and name Strings (java.lang.String).
- The E-mail eWay does not currently support sending or receiving messages containing multi-level nested MIME parts.

5.3 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 Java Composite Application Platform Suite Installer's Documentation tab. The **EmaileWay_Sample.zip** file contains the various sample Project zip files. 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 (for this sample, **prjEmail_JCD.zip**) and click **Import**. After the sample Project is successfully imported, click **Close**.
5. Before an imported sample Project can be run you must do the following:
   - Create an **Environment** (see **"Creating an Environment" on page 70**)
   - Configure the eWays for your specific system (see **"Configuring the eWays" on page 71**)
   - Create a **Deployment Profile** (see **"Creating and Activating the Deployment Profile" on page 73**)

- Create and start a domain (see **"Creating and Starting the Domain" on page 76**)
- Build and deploy the Project (see **"Building and Deploying the Project" on page 76**)
- The following pages provide step by step directions for manually creating the **prjEmail_JCD** Project.

5.4 The prjEmail_JCD Project Overview

The e-Mail eWay Project, **prjEmail_JCD**, demonstrates the following:

Send Message

- The inbound File eWay subscribes to an external directory. When a message is present, the File eWay picks up the message (an XML file containing the ToAddress, CCAddress, FromAddress, subject, and message text) and publishes the message to the jcdSend Collaboration.
- The jcdSend Collaboration formats the ToAddress, CCAddress, FromAddress, subject, and message, and publishes the message to the outbound e-Mail eWay.
- The outbound e-Mail eWay publishes the e-mail message to a SMTP server.

Receive Message

- The inbound e-Mail eWay subscribes to an external e-mail account, polling the account every 5 seconds. When e-mail is present, the e-Mail eWay publishes the message to the jcdReceive Collaboration.
- The jcdReceive Collaboration parses the email message, and writes only the message text to a text file.
- The outbound File eWay writes the message text to an external directory.

Read Message

- The inbound File eWay, acting as a trigger, subscribes to an external directory. When a message is present, the File triggers the jcdRead Collaboration.
- The e-Mail eWay, triggered by the jcdRead Collaboration, subscribes to the email server. If a message is present, the e-Mail eWay publishes the message to the jcdRead Collaboration.
- The jcdRead Collaboration parses the ToAddress, CCAddress, FromAddress, subject, and message.
- The outbound File eWay writes this information to an external directory.

Receive Message checks for e-mail continuously, while **Read Message** only checks for e-mail when the Business Process is triggered by a file.

5.5 Creating the prjEmail_JCD Project

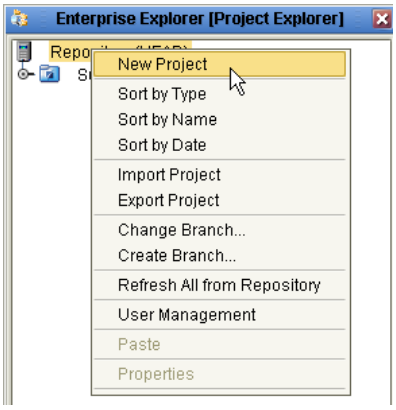
The following pages provide step by step directions for manually creating the sample Project components.

5.5.1 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** (see Figure 24). A new Project (**Project1**) appears on the Project Explorer tree.

Figure 24 Enterprise Explorer - New Project



3. Rename the Project (for this sample, **prjEmail_JCD**).

5.5.2 Creating a Connectivity Maps

The Connectivity Map provides a canvas for assembling and configuring a Project's components.

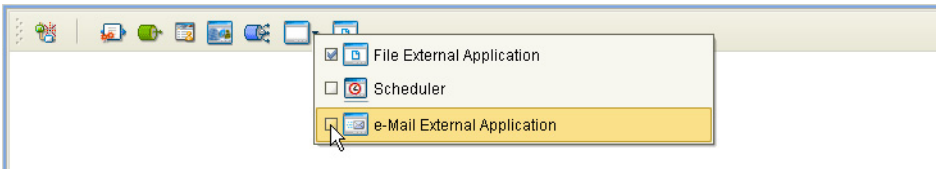
1. From the Project Explorer tree, right-click the new **prjEmail_JCD** Project and select **New > Connectivity Map** from the shortcut menu.
2. The New Connectivity Map appears and a node for the Connectivity Map is added under the Project on the Project Explorer tree labeled **CMap1**. Rename the Connectivity Map **cmSend**.
3. Create two additional Connectivity Maps named **cmRead** and **cmReceive**.

The icons in the toolbar represent the available components used to populate the Connectivity Map canvas.

Selecting the External Applications

In the Connectivity Maps, eWays are associated with External Systems. For example, to establish a connection to an external e-Mail application, you must first select e-Mail as an External Application to use in your Connectivity Map (see Figure 25).

Figure 25 Connectivity Map - External Applications



1 Click the **External Application** icon on the Connectivity Map toolbar,

2 Select the External Applications you require to create your Project (for this sample, **File** and **e-Mail**). Icons representing the selected External Applications are added to the Connectivity Map toolbar.

Populating the Connectivity Maps

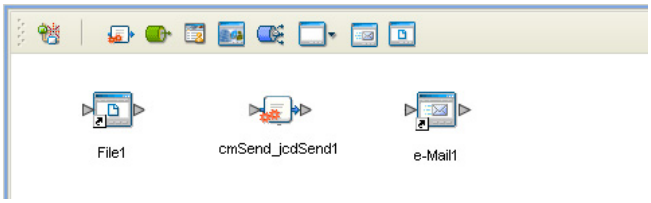
For the **cmSend** Connectivity Map, add the Project components by dragging the icons from the toolbar to the canvas.

1 For this sample, drag the following components onto the Connectivity Map canvas as displayed in Figure 26.

- e-Mail External System
- Service (A service is a container for Java Collaborations, Business Processes, eTL processes, and so forth)
- File External System

2 Rename the cmSend_Service1 to **cmSend_jcdSend1** (see Figure 26):

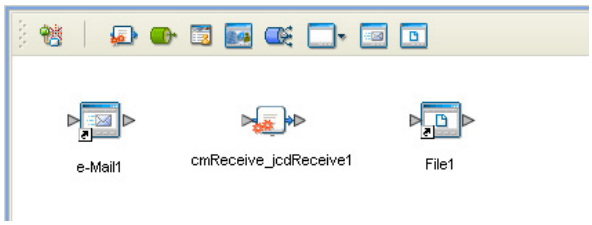
Figure 26 cmSend Connectivity Map with Components



3 Populate the **cmReceive** Connectivity Map with the following components, as displayed in Figure 27:

- File External System
- Service (rename to **cmReceive_jcdReceive1**)
- e-Mail External System

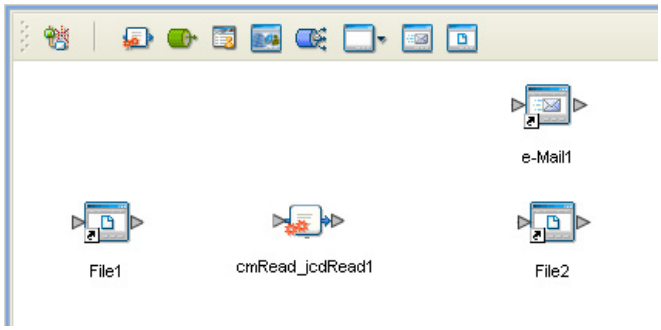
Figure 27 cmReceive Connectivity Map



4 Populate the **cmRead** Connectivity Map with the following components, as displayed in Figure 28:

- File External System (2)
- Service (rename to **cmRead_jcdRead1**)
- e-Mail External System

Figure 28 cmRead Connectivity Map



5 Save your current changes to the Repository.

5.5.3 Creating an OTD Using the Object Type Definition Wizard

The prjEmail_JCD Project contains three Java Collaborations that utilize a user-created Object Type Definition (OTD). This OTD is created from a Document Type Definition (DTD) using the Object Type Definition Wizard. A DTD file, eMail.dtd, is provided with the sample Project file. To create the **eMail_SendEmail** OTD, do the following:

1 From the Project Explorer, right-click the **prjEmail_JCD** Project and select New > Object Type Definition. The Object Type Definition Wizard appears.

2 For step 1 of the wizard, from the Select Wizard Type window, select **DTD**, and click **Next**.

3 For step 2 of the wizard, from the Browse DTD Files window, locate and select the **eMail.dtd** file that you downloaded to your computer with the e-Mail eWay's sample Projects. Click **Select** and click **Next**.

4 For step 3 of the wizard, from the Select Document Elements field, make sure that **eMail_SendEmail** is selected and click **Next**.

5 For step 4 of the wizard, keep all of the default OTD options, and click **Finish**. The OTD Editor appears with the new **eMail_SendEmail** OTD, and the OTD is added to the prjEmail_JCD Project.

5.5.4 Creating the Collaboration Definitions

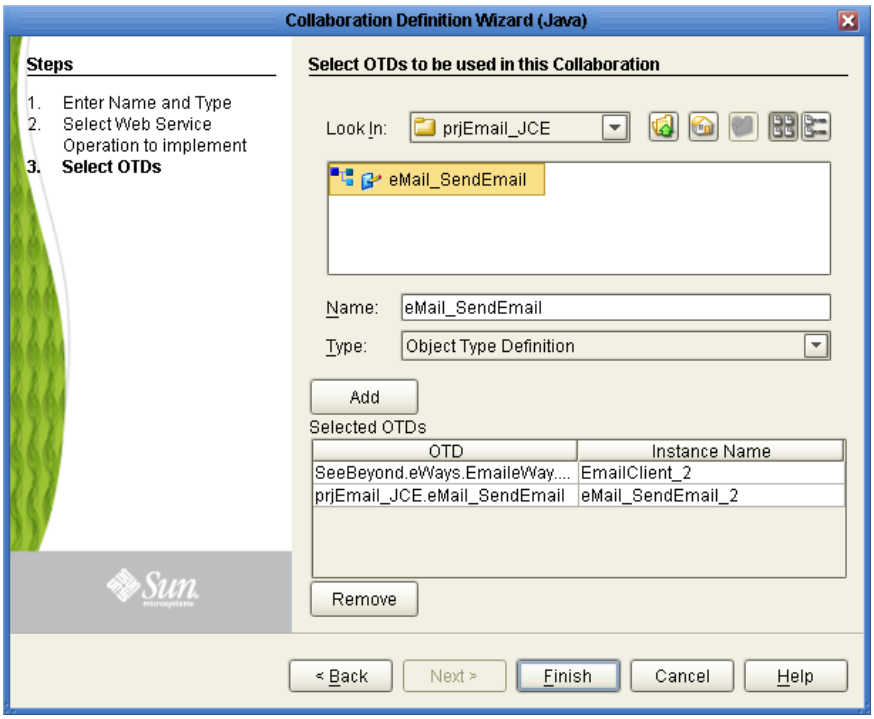
The next step in the sample is to create three Java Collaborations using the Collaboration Definition Wizard (Java). Once the Collaboration Definitions have been created, the Business Rules of the Collaborations can be written using the Collaboration Editor (Java).

Creating the jcdSend Collaboration Definition

The **jcdSend** Collaboration defines transactions between the inbound File eWay, the e-Mail eWay, and the e-Mail External Application.

1. From the Project Explorer, right-click the sample Project and select **New > Collaboration Editor (Java)** from the shortcut menu. The **Collaboration Definition Wizard (Java)** appears.
2. Enter a Collaboration Definition name (for this sample **jcdSend**) and click **Next**.
3. For Step 2 or the wizard, from the Web Services Interfaces selection window, double-click **Sun SeeBeyond** > **eWays** > **File** > **FileClient** > **receive**. The File Name field now displays **receive.** Click **Next**.
4. For Step 3 of the wizard, from the Select OTDs to be used in this Collaboration window, double-click **Sun SeeBeyond** > **eWays** > **EmaileWay** > **EmailClient**. The **EmailClient** OTD is added to the Selected OTDs field.
5. Click the **Up One Level** button to return your Repository. From the Select OTDs to be used in this Collaboration window, double-click **prjEmail_JCD** > **eMail_SendEmail**. The **eMail_SendEmail** OTD is added to the Selected OTDs field (see Figure 29).

Figure 29 Collaboration Definition Wizard (Java)



6 Click **Finish**. The Collaboration Editor (Java) with the new jcdSend Collaboration appears in the right pane of the Enterprise Designer.

Creating the jcdReceive Java Collaboration

The **jcdReceive** Collaboration (Java) defines transactions between the inbound e-Mail eWay application to the Outbound File eWay.

1 From the Project Explorer, right-click the sample Project and select **New > Collaboration Editor (Java)** from the shortcut menu. The **Collaboration Definition Wizard (Java)** appears.

2 Enter a Collaboration Definition name (for this sample **jcdReceive**) and click **Next**.

3 For Step 2 of the wizard, from the Web Services Interfaces selection window, double-click **Sun SeeBeyond** > **eWays** > **EmaileWay** > **EmailClient** > **receive**. The Name field now displays **receive.** Click **Next**.

4 For Step 3, **Select OTDs**, from the Select OTDs selection window, double-click **Sun SeeBeyond** > **eWays** > **File** > **FileClient**. The **FileClient** OTD is added to the Selected OTDs field.

5 Click the **Up One Level** button to return your Repository. From the Select OTDs to be used in this Collaboration window, double-click **prjEmail_JCD** > **eMail_SendEmail**. The **eMail_SendEmail** OTD is added to the Selected OTDs field.

6 Click **Finish**. The Collaboration Editor with the new **jcdReceive** Collaboration appears.

Creating the jcdRead Java Collaboration

The **jcdRead** Collaboration (Java) defines transactions made between the inbound File eWay, the e-Mail eWay, and the Outbound File eWay.

1. From the Project Explorer, right-click the sample Project and select **New > Collaboration Editor (Java)** from the shortcut menu. The **Collaboration Definition Wizard (Java)** appears.
2. Enter a Collaboration Definition name (for this sample **jcdRead**) and click **Next**.
3. For Step 2 of the wizard, from the Web Services Interfaces selection window, double-click **Sun SeeBeyond** > **eWays** > **EmaileWay** > **EmailClient** > **receive**. The Name field now displays **receive.** Click **Next**.
4. For Step 3, **Select OTDs**, from the Select OTDs selection window, double-click **Sun SeeBeyond** > **eWays** > **File** > **FileClient**. The Selected OTDs field now lists the **FileClient** OTD.
5. Click the **Up One Level** button to return your Repository. From the Select OTDs selection window, double-click **Sun SeeBeyond** > **eWays** > **File** > **FileClient**. The **FileClient** OTD is added to the Selected OTDs field.
6. Click the **Up One Level** button to return your Repository. From the Select OTDs to be used in this Collaboration window, double-click **prjEmail_JCD** > **eMail_SendEmail**. The **eMail_SendEmail** OTD is added to the Selected OTDs field.
7. Click **Finish**. The Collaboration Editor with the new **jcdRead** Collaboration appears.

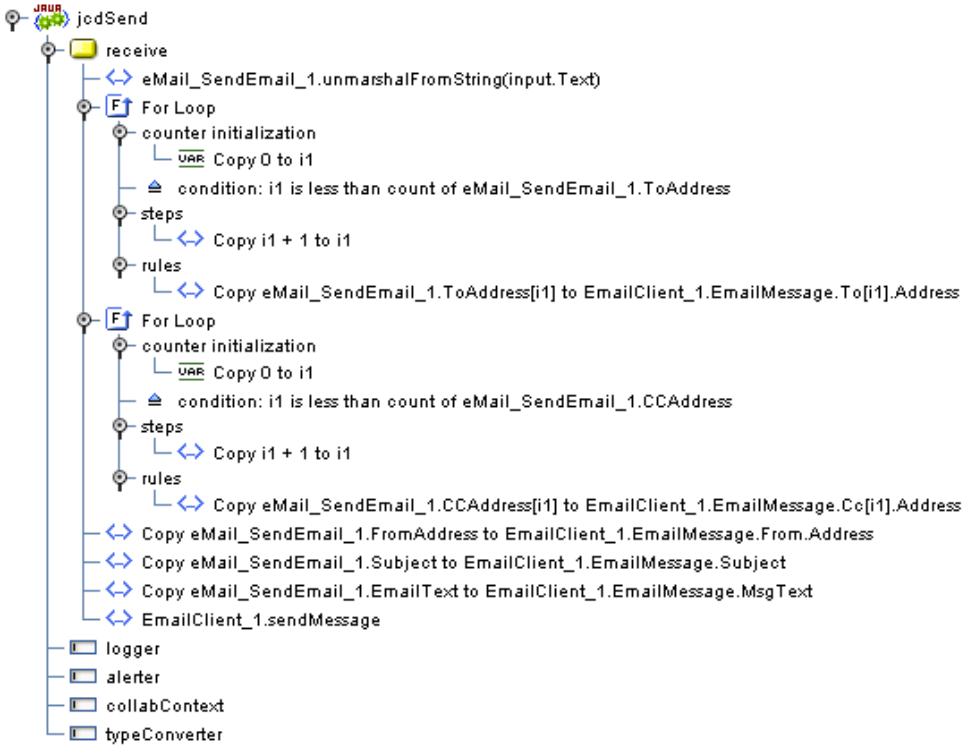
5.5.5 Using the Collaboration Editor (Java)

The next step in the sample is to create the Business Rules of the Collaborations using the Collaboration Editor (Java).

Creating the jcdSend Business Rules

The **jcdSend** Collaboration contains the Business Rule displayed in Figure 30.

Figure 30 jcdSend Collaboration Business Rules

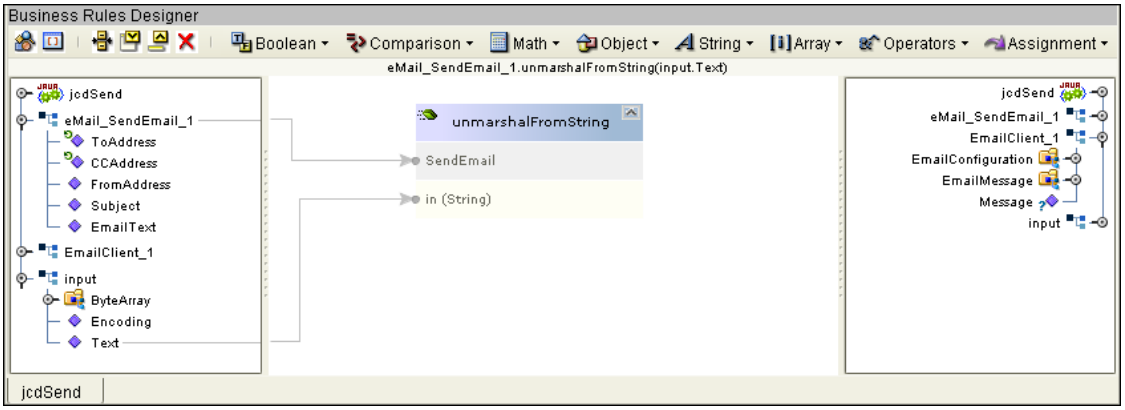


To create the **jcdSend** Collaboration Business Rules do the following:

1. From the Project Explorer tree, double-click **jcdSend** to open the Collaboration Editor (Java) to the **jcdSend** Collaboration.
2. The imported sample includes comments with the Business Rules. To create comments for the Business Rules, click the comment icon on the Business Rules toolbar. The **Enter a Comment** dialog box appears. Enter the comment and click **OK**. The comment is placed on the Business Rules tree under the last selected item. Once the Comment is created, it can be moved by clicking the comment and dragging it up or down the Business Rules tree to a new location.
3. Create the **eMail_SendEmail_1.unmarshalFromString(input.Text)** rule:
   - D From the left pane of the Business Rules Designer, right-click **eMail_SendEmail_1** and click **Select method to call,** from the shortcut menu.
   - E From the method selection window, select **unmarshalFromString(String in)**. The **unmarshalFromString** method box appears.

F Map **Text** under **input** in the left pane of the Business Rules Designer, to the **in (String)** input node of the **unmarshalFromString** method box. To do this, click on **Text** in the left pane of the Business Rules Designer, and drag your cursor to the **in (String)** input node of the **unmarshalFromString** method box. A link now connects the two nodes (see Figure 31).

Figure 31 jcdSend - Business Rules Designer

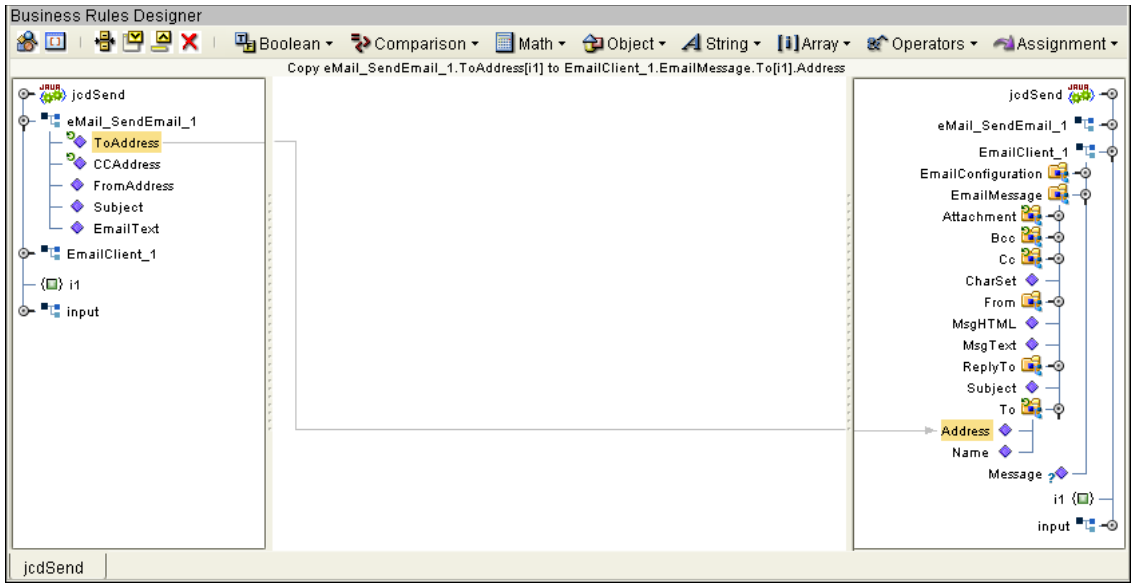


4 Create the **For Loop** with the **Copy eMail_SendEmail_1.ToAddress[i1] to EmailClient_1.EmailMessage.To[i1].Address** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B Map **ToAddress**, under **eMail_SendEmail_1** in the left pane of the Business Rules Designer, to **Address** under **EmailClient_1 > EmailMessage > To**, in the right pane of the Business Rules Designer. The For Loop with the new Business Rule on the Business Rules tree (see Figure 32).

Figure 32 jcdSend - Business Rules Designer



5 Create the **For Loop** with the **Copy eMail_SendEmail_1.ToAddress[i1] to EmailClient_1.EmailMessage.To[i1].Address** rule:

A Collapse the **For Loop** that you created above. With the **For Loop** on the Business Rules tree selected, click the **rule** icon on the Business Rules toolbar, to add a new rule.

B Map **CCAddress**, under **eMail_SendEmail_1** in the left pane of the Business Rules Designer, to **Address** under **EmailClient_1 > EmailMessage > Cc**, in the right pane of the Business Rules Designer. The For Loop with the new Business Rule on the Business Rules tree

6 Create the **Copy eMail_SendEmail_1.Subject to EmailClient_1.EmailMessage.Subject** rule:

A Collapse the **For Loop** that you created above. With the **For Loop** on the Business Rules tree selected, click the **rule** icon on the Business Rules toolbar, to add a new rule.

B Map **FromAddress**, under **eMail_SendEmail_1** in the left pane of the Business Rules Designer, to **Address** under **EmailClient_1 > EmailMessage > From**, in the right pane of the Business Rules Designer.

7 Create the **Copy eMail_SendEmail_1.FromAddress to EmailClient_1.EmailMessage.From.Address** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B Map **Subject**, under **eMail_SendEmail_1** in the left pane of the Business Rules Designer, to **Subject** under **EmailClient_1 > EmailMessage**, in the right pane of the Business Rules Designer.

8 Create the **Copy eMail_SendEmail_1.EmailText to EmailClient_1.EmailMessage.MsgText** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B Map **EmailText**, under **eMail_SendEmail_1** in the left pane of the Business Rules Designer, to **MsgText** under **EmailClient_1 > EmailMessage**, in the right pane of the Business Rules Designer.

9 Create the **EmailClient_1.sendMessage** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B From the left pane of the Business Rules Designer, right-click **EmailClient_1** and click **Select method to call,** from the shortcut menu.

C From the method selection window, select **sendMessage()**. The **sendMessage** method box appears.

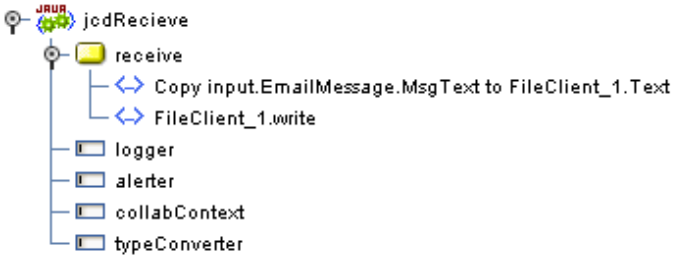
10 From the editor's toolbar, click **Validate** to check the Collaboration for errors.

11 Save your current changes to the repository.

Creating the jcdReceive Collaboration Business Rules

The **jcdReceive** Collaboration contains the Business Rule displayed in Figure 33

Figure 33 jcdReceive Collaboration Business Rules



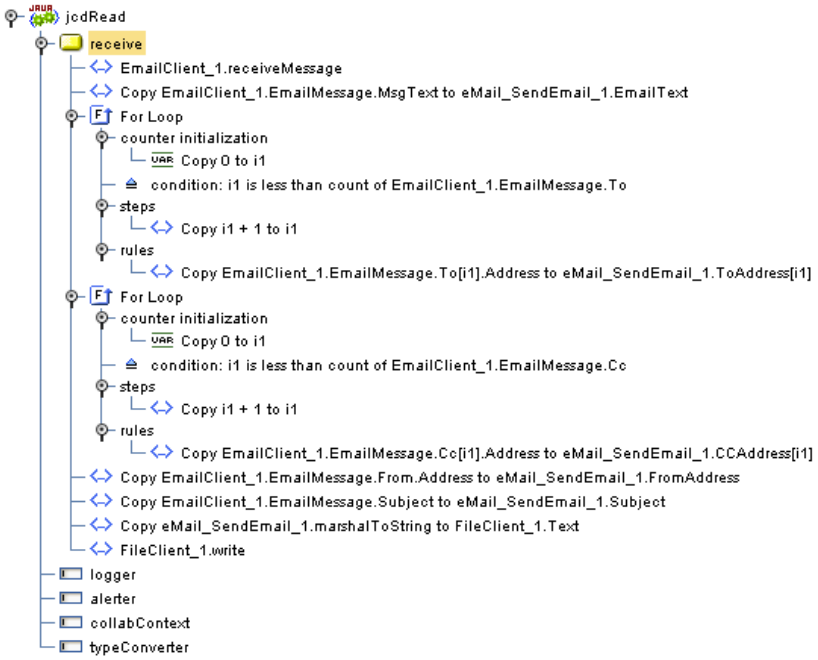
To create the **jcdReceive** Collaboration Business Rules do the following:

1. From the Project Explorer tree, double-click **jcdReceive** to open the Collaboration Editor (Java) to the **jcdReceive** Collaboration.
2. Create the **Copy input.EmailMessage.From.Address to FileClient_1.Text** rule:
   - A From the Business Rules toolbar click the **rule** icon to add a new rule.
   - B Map **msgText** under **input** > **EmailMessage** in the left pane of the Business Rules Designer, to **Text** under **FileClient_1** in the right pane of the Business Rules Designer.
3. Create the **FileClient_1.write** Business Rule:
   - A From the Business Rules toolbar click the **rule** icon to add a new rule.
   - B From the left pane of the Transformation Designer, right-click **FileClient_1** and click **Select a method to call** from the shortcut menu. The method selection box appears.
   - C Select **write()**. The **write** method box appears in the Business Rules Designer canvas.
4. From the editor's toolbar, click **Validate** to check the Collaboration for errors.
5. Save your current changes to the repository.

Creating the jcdRead Collaboration Business Rules

The **jcdRead** Collaboration contains the Business Rule displayed in Figure 33

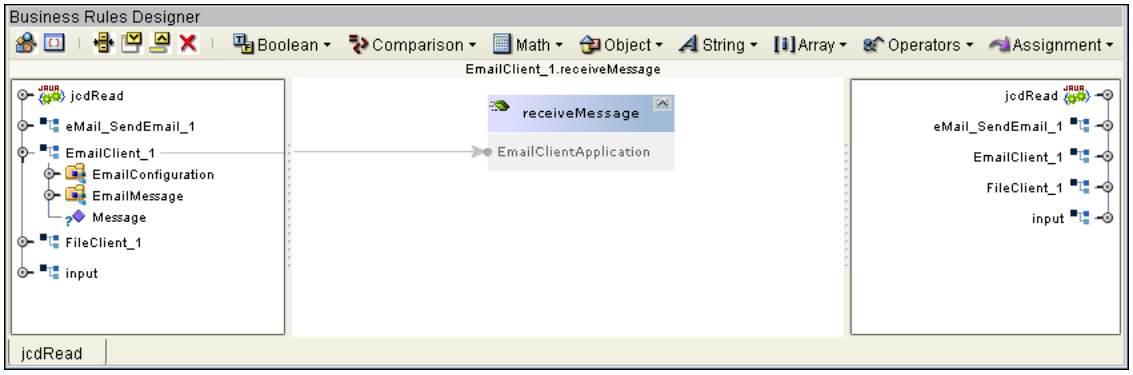
Figure 34 jcdRead Collaboration Business Rules



To create the **jcdRead** Collaboration Business Rules do the following:

1. From the Project Explorer tree, double-click **jcdRead** to open the Collaboration Editor (Java) to the **jcdRead** Collaboration.
2. Create the **Copy input.EmailMessage.From.Address to FileClient_1.Text** rule:
   - A. From the Business Rules toolbar click the **rule** icon to add a new rule.
   - B. From the left pane of the Transformation Designer, right-click **EmailClient_1** and click **Select a method to call** from the shortcut menu. The method selection box appears.
   - C. Select **receiveMessage()**. The **receiveMessage** method box appears in the Business Rules Designer canvas (see Figure 35).

Figure 35 jcdRead - Business Rules Designer



3. Create the **Copy EmailClient_1.EmailMessage.MsgText to eMail_SendEmail_1.EmailText** rule:

A From the Business Rules toolbar click the **rule** icon to add a new rule.

B Map **MsgText**, under **EmailClient_1 > EmailMessage** in the left pane of the Business Rules Designer, to **EmailText**, under **eMail_SendEmail_1** in the right pane of the Business Rules Designer.

4 Create the **For Loop** with the **Copy EmailClient_1.EmailMessage.To[i1].Address to eMail_SendEmail_1.ToAddress[i1]** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B Map **ToAddress**, under **eMailClient_1 > EmailMessage** in the left pane of the Business Rules Designer, to **ToAddress** under **eMail_SendEmail_1** in the right pane of the Business Rules Designer. The For Loop with the new Business Rule on the Business Rules tree.

5 Create the **For Loop** with the **Copy EmailClient_1.EmailMessage.Cc[i1].Address to eMail_SendEmail_1.CCAddress[i1]** rule:

A Collapse the **For Loop** that you created above. With the **For Loop** on the Business Rules tree selected, click the **rule** icon on the Business Rules toolbar, to add a new rule.

B Map **Address**, under **eMailClient_1 > EmailMessage** in the left pane of the Business Rules Designer, to **CCAddress** under **eMail_SendEmail_1** in the right pane of the Business Rules Designer. The For Loop with the new Business Rule on the Business Rules tree

6 Create the **Copy eMail_SendEmail_1.Subject to EmailClient_1.EmailMessage.Subject** rule:

A Collapse the **For Loop** that you created above. With the **For Loop** on the Business Rules tree selected, click the **rule** icon on the Business Rules toolbar, to add a new rule.

B Map **Address**, under **EmailClient_1 > EmailMessage > From** in the left pane of the Business Rules Designer, to **FromAddress** under **eMail_SendEmail_1** in the right pane of the Business Rules Designer.

7 Create the **Copy EmailClient_1.EmailMessage.Subject to eMail_SendEmail_1.Subject** rule:

A From the Business Rules toolbar, click the **rule** icon to add a new rule.

B Map **Subject**, under **EmailClient_1 > EmailMessage** in the left pane of the Business Rules Designer, to **Subject** under **eMail_SendEmail_1** in the right pane of the Business Rules Designer.

8 Create the **Copy eMail_SendEmail_1.marshalToString to FileClient_1.Text** rule:

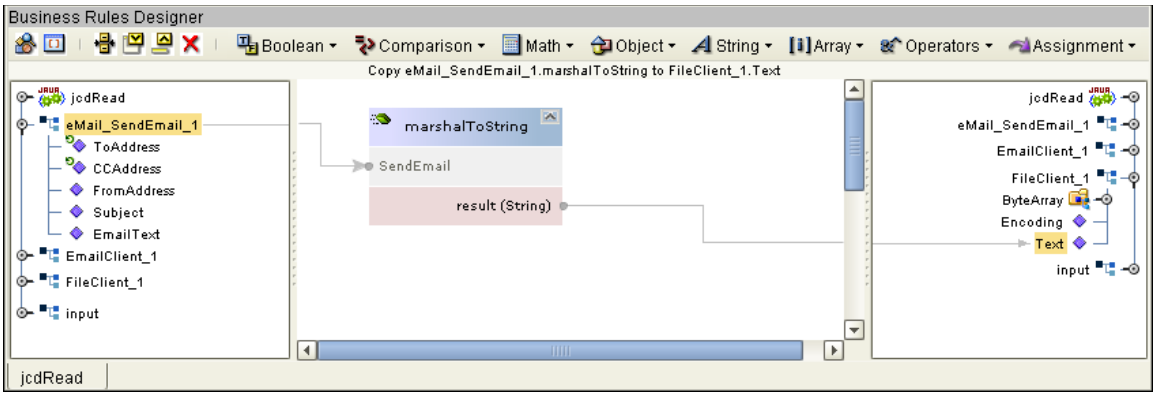
A From the Business Rules toolbar click the **rule** icon to add a new rule.

B From the left pane of the Transformation Designer, right-click **eMail_SendEmail_1** and click **Select a method to call** from the shortcut menu. The method selection box appears.

C Select **marshalToString()**. The **marshalToString** method box appears in the Business Rules Designer canvas.

D Map the **result(String)** output node of the **marshalToString** method box, to **Text**, under **FileClient_1** in the right pane of the Business Rules Designer (see Figure 36).

Figure 36 jcdRead - Business Rules Designer



9 Create the **FileClient_1.write** rule:

A From the Business Rules toolbar click the **rule** icon to add a new rule.

B From the left pane of the Transformation Designer, right-click **FileClient_1** and click **Select a method to call** from the shortcut menu. The method selection box appears.

C Select **write()**. The **write** method box appears in the Business Rules Designer canvas.

10 From the editor's toolbar, click **Validate** to check the Collaboration for errors.

11 Save your current changes to the repository.

For more information on the Collaboration Rules Editor see the *Sun SeeBeyond eGate™ Integrator User's Guide*.

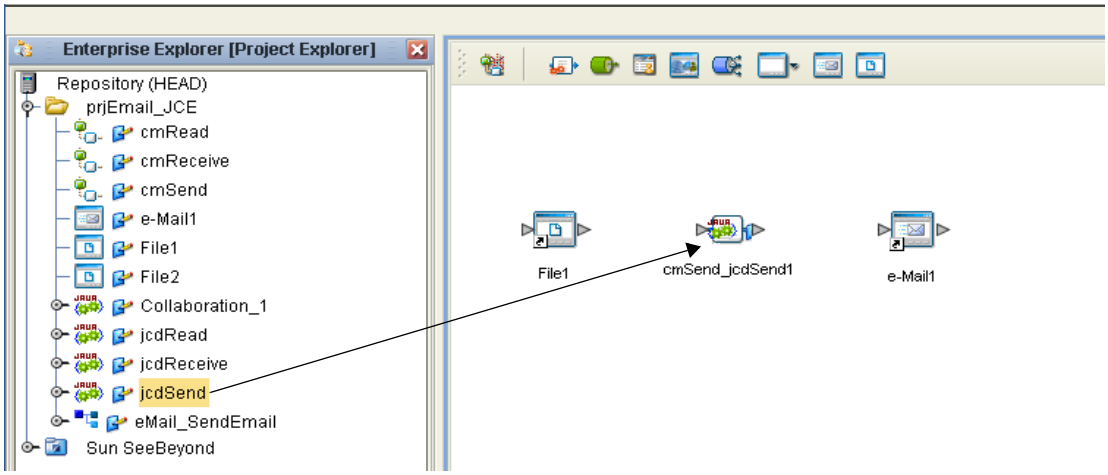
5.5.6 Binding the eWay Components

After the Collaborations have been written, the components are associated and bindings are created in the Connectivity Maps.

1 From the Project Explorer, double-click **cmSend** to display the **cmSend** Connectivity Map.

2 Drag and drop the **jcdSend** Collaboration from the Project Explorer to the **cmSend_jcdSend1** Service. If the Collaboration was successfully associated, the Service's "gears" icon changes from red to green (see **Figure 37 on page 69**).

Figure 37 Connectivity Map - Binding the Components

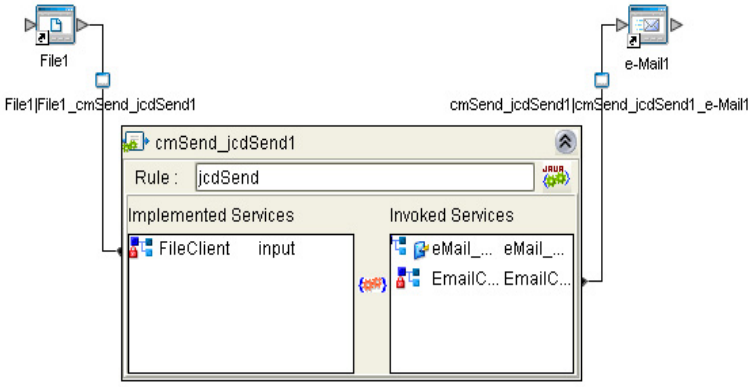


3 From the Connectivity Map canvas, double-click **cmSend_jcdSend1**. The **cmSend_jcdSend1** binding dialog box appears using the **jcdSend** Rule.

4 From the **cmSend_jcdSend1** binding dialog box, map **FileClient Input** (under Implemented Services) to the output node of the inbound **File1** External Application. To do this, click on **FileClient Input** under Implemented Services in the cmSend_jcdSend1 binding box, and drag your cursor to the output node of the **File1** External Application. A link now joins the two components.

5 From the **cmSend_jcdSend1** binding dialog box, map **EmailClient** (under Invoked Services) to the input node of the **e-Mail1** External Application.

Figure 38 Connectivity Map - Associating (Binding) the Project's Components



6 Minimize the **cmSend_jcdSend1** binding dialog box.

7 From the Project Explorer, double-click **cmReceive** to display the **cmReceive** Connectivity Map.

8 Drag and drop the **jcdReceive** Collaboration from the Project Explorer to the **cmReceive_jcdReceive1** Service.

9 From the Connectivity Map, double-click the **cmReceive_jcdReceive1** Service. The **cmReceive_jcdReceive1** binding dialog box appears with the **jcdReceive** Rule.

10 From the **cmReceive_jcdReceive1** binding dialog box, map **EmailClient input** (under Implemented Services) to output node of the **e-Mail1** External Application.

11 From the **cmReceive_jcdReceive1** binding dialog box, map **FileClient_1, FileClient** (under Invoked Services) to the input node of the outbound **File1** External Application.

12 Minimize the **cmReceive_jcdReceive1** binding dialog box.

13 From the Project Explorer, double-click **cmRead** to display the **cmRead** Connectivity Map.

14 From the Connectivity Map, double-click the **cmRead_jcdRead1** Service. The **cmRead_jcdRead1** binding dialog box appears with the **jcdRead** Rule.

15 From the **cmRead_jcdRead1** binding dialog box, map **FileClient input** (under Implemented Services) to the output node of the **File1** External Application.

16 From the **cmRead_jcdRead1** binding dialog box, map **EmailClient_1 EmailClient** (under Invoked Services) to the input node of the **e-Mail1** External Application.

17 From the **cmRead_jcdRead1** binding dialog box, map **FileClient_1 FileClient** (under Invoked Services) to the input node of the outbound **File2** External Application.

18 Minimize the **cmRead_jcdRead1** binding dialog box, and save all your current changes to the Repository.

5.5.7 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 Explorer and 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 **envEmail_JCD**.

4 Right-click **envEmail_JCD** and select **New > e-Mail External System**. Name the External System **esEmail** and click **OK**. **esEmail** is added to the Environment Editor.

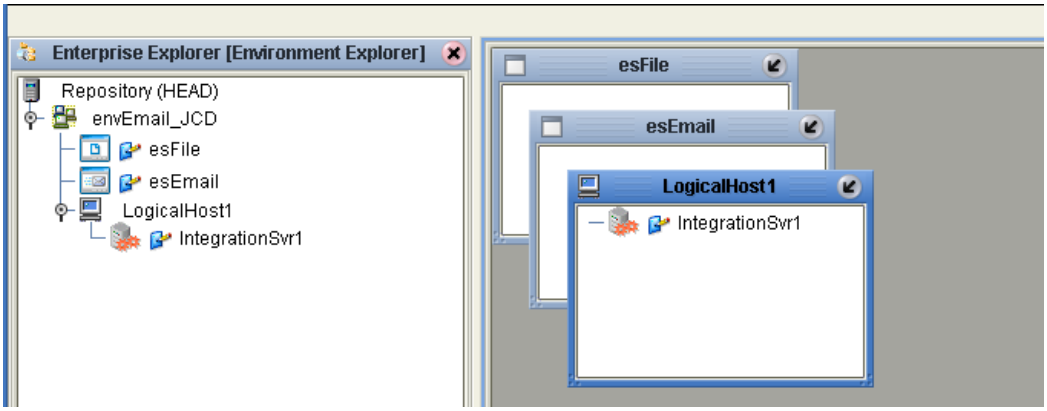
5 Right-click **envEmail_JCD** and select **New > File External System**. Name the External System **esFile** and click **OK**. **esFile** is added to the Environment Editor.

6 Right-click **envEmail_JCD** 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 39 on page 71**.

Figure 39 Environment Editor - envEmail_JCD

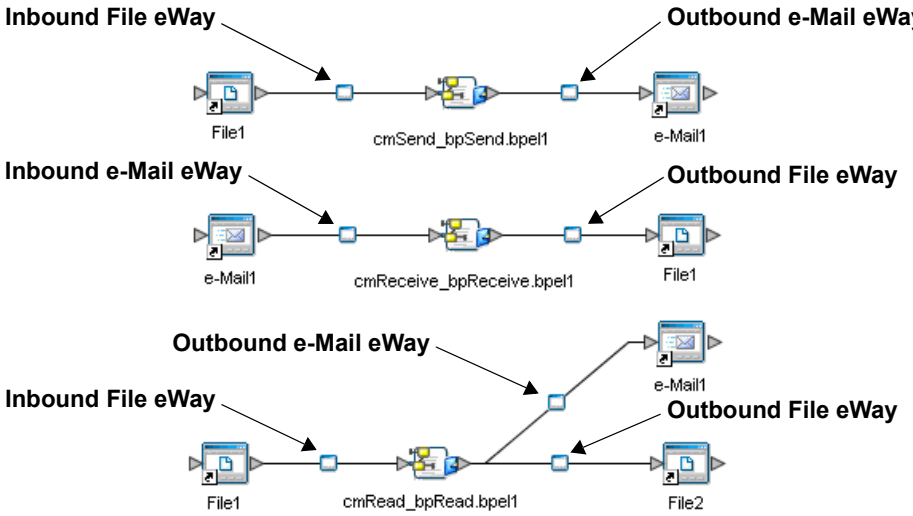


9 Save your current changes to the Repository.

5.5.8 Configuring the eWays

The prjEmail_JCD Project uses four eWays, each represented in the Connectivity Maps as a node between an External Application and a Service. eWays facilitate communication and movement of data between the external applications and the eGate system.

Figure 40 eWays



The eWay properties are set from both the Project Explorer's Connectivity Maps and the Environment Explorer tree.

Configuring the File eWay Properties

1 From the cmSend Connectivity Map, double-click the inbound **File1** eWay (see Figure 40). 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 16, and click **OK**.

Table 16 jcdSend - Inbound File eWay Settings

Inbound File eWay Connection Parameters	
Input file name	email.xml

3 In the same way, modify the properties of the cmRead Connectivity Map's inbound **File1** eWay, entering **trigger.txt** as the Input file name property value.

4 From the cmReceive Connectivity Map, modify the outbound **File1** eWay properties for your system, including the settings in Table 17.

Table 17 Outbound File eWay Settings

Outbound File eWay Connection Parameters	
Output file name	output%d.dat

5 In the same way, modify the properties of the cmRead Connectivity Map's outbound **File2** eWay.

6 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.

7 Modify the File eWay Environment properties for your system, including the settings in Table 18, and click **OK**.

Table 18 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:/temp*
Outbound File eWay > Parameter Settings Set as directed, otherwise use the default settings	
Directory	*Select a directory, for example C:/temp*

Configuring the e-Mail eWay Properties

1 The default e-Mail eWay **Connectivity Map** properties are sufficient for this sample.

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

3 Modify the e-Mail eWay Environment properties for your system, including the settings in Table 19.

Table 19 e-Mail eWay Environment Explorer Properties

e-Mail eWay Environment Properties	
Inbound Email eWay > Connection Settings Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Port Receive	*TCP/IP Port Number for receiving e-mail*
User Receive	*User name for receiving e-mail*
Password Receive	*User password for receiving e-mail*
Outbound Email eWay > Connection Settings > Send SMTP Set as directed, otherwise use the default settings	
Host Send	*Host name of the sending SMTP server*
Port Send	*TCP/IP Port Number used for sending e-mail*
User Send	*User name for sending e-mail*
Password Send	*The password for sending email messages.*
Outbound Email eWay > Connection Settings > Receive POP3 Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Port Receive	*TCP/IP Port Number used for receiving e-mail*
User Receive	*User name for receiving e-mail*
Password Receive	*The password for receiving email messages.*
Session Authentication	*NO*

5.5.9 Configuring the Integration Server

You must set your Sun 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. Enter **STC** as the **Specific Value** and as the **Confirm Password**, and click **OK**.
4. 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*.

5.5.10 Creating and Activating the Deployment Profile

Deployment Profiles are specific instances of a Project in a particular Environment. A Deployment Profile contains information regarding the assignment of services and

message destinations to integration and message servers (JMS IQ Managers). It also contains version information for all versionable objects in the Project. Deployment Profiles are created using the Deployment Editor.

The prjEmail_JCD Project performs multiple operations and requires two different Deployment Profiles; **dp_jcdRead**, and **dp_jcdReceive**. This allows part of the Project to be undeployed.

Create the dp_jcdRead Deployment Project

To create the dp_jcdRead Deployment Profile do the following:

1. From the Project Explorer, right-click **prjEmail_JCD** and select **New** > **Deployment Profile** from the shortcut menu.
2. Enter **dp_jcdRead** as the name for the Deployment Profile. Make sure that the selected Environment is **envEmail_JCD**. Click **OK**. The Deployment Profile Editor appears.
3. At this point, for most Project, you would click the Automap button and the components would be automatically mapped to the correct external system. For this Project you must map the components manually so that the Projects operations are split between two Deployment Profiles. Map the components (drag-and-drop) to the external systems as follows:

   esEmail:

   - cmRead_jcdRead1 -> e-Mail1
   - cmSend_jcdSend1 -> e-Mail1

   esFile:

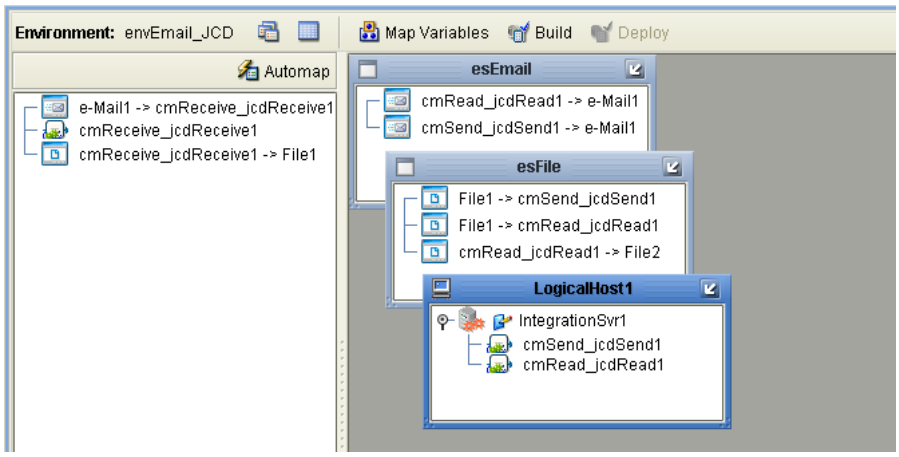
   - File1 -> cmSend_jcdSend1
   - File1 -> cmRead_jcdRead1
   - cmRead_jcdRead1 -> File2

   LogicalHost1:

   - cmSend_jcdSend1
   - cmRead_jcdRead1

   The Deployment Profile should now appear as displayed in **Figure 41 on page 75**.

Figure 41 dp_jcdRead Deployment Profile



Create the dp_jcdReceive Deployment Project

To create the dp_jcdReceive Deployment Profile do the following:

1. From the Project Explorer, right-click **prjEmail_JCD** and select **New** > **Deployment Profile** from the shortcut menu.
2. Enter **dp_jcdReceive** as the name for the Deployment Profile. Make sure that the selected Environment is **envEmail_JCD**. Click **OK**. The Deployment Profile Editor appears.
3. Map the components (drag-and-drop) to the external systems as follows:

   esEmail:

   - e-Mail -> cmReceive1

   esFile:

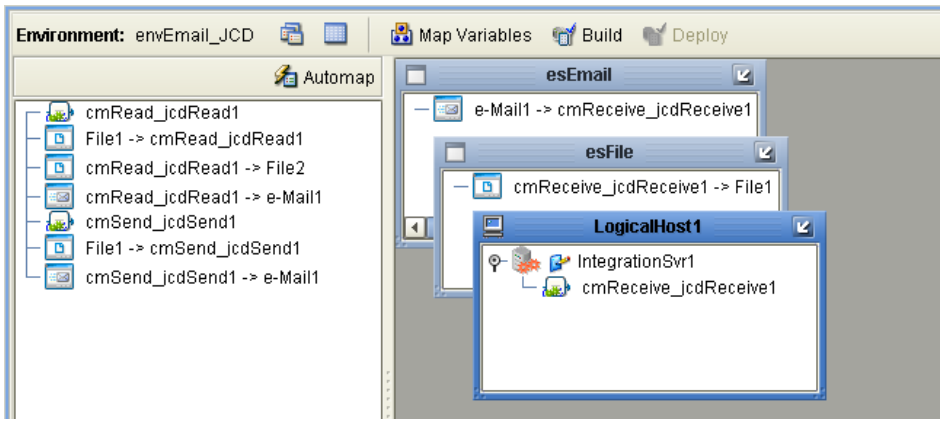
   - cmReceive_jcdReceive1 -> File1

   LogicalHost1:

   - cmReceive_jcdReceive1

   The Deployment Profile should now appear as displayed in **Figure 42 on page 76**.

Figure 42 dp_jcdReceive Deployment Profile



4 Save your changes to the Repository.

5.5.11 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 Integration 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.

5.5.12 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.*

5.5.13 Running the Sample

The prjEmail_JCD Project Demonstrates three different operations (see **"The prjEmail_JCD Project Overview" on page 55**).

To run the **Send Message** and **Read Message** operations, the **dp_jcdReceive** Deployment Profile must be undeployed from the Sun SeeBeyond Enterprise Manager.

To run the **Send Message** and **Receive Message** operations, the **dp_jcdRead** and **dp_jcdReceive** Deployment Profiles are deployed, and the **Receive Message** operation supersedes the **Read Message** operation.

For information on using the Sun SeeBeyond Enterprise Manager see the *Sun SeeBeyond eGate™ Integrator System Administration Guide*.

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.

5.6 Enabling Japanese Character Support

The e-Mail eWay supports Japanese character encoding in both the header and message content of the e-mail message (for both text/plain and text/HTML). The e-Mail eWay conforms to RFC2047 standards for Multipurpose Internet Mail Extensions (MIME).

To enable Japanese character encoding for outbound messages, the e-Mail eWay Connectivity Map properties, **Text encoding** and **Header encoding,** must be configured as follows:

1. From the e-Mail eWay Connectivity Map Properties Editor, navigate to the **Configuration > Outbound Email eWay > Connection Settings > Send SMTP** properties.
2. Set the Text encoding property value to **iso-2022-jp** (Japanese).
3. Verify that the Header encoding property value is set to the default value **B**.

For inbound e-Mail messages, encoding is detected automatically. There is no need to configure encoding for the inbound eWay settings.

Japanese Character Support in the EmailMessage OTD

The **EmailMessage** OTD is the user interface for the e-Mail header. When configured for Japanese character support the EmailMessage OTD operates as follows:

- The e-Mail header (String) fields, To/cc/Bcc, subject, and attachment name, use iso-2022-jp (Japanese) encoding. These fields are set using the following methods:
  - **EmailAddress.SetName(String)** for To/cc/Bcc of the EmailMessage
  - **EmailMessage.setSubject(String)**
  - EmailAttachment.setName(String)
- Outbound messages use iso-2022-jp (Japanese) encoding for message content. Message content us set using one of the following methods:
  - **EmailMessage.setMsgText(String)**
  - **EmailMessage.setHTML(String)**
- The attachment data set by the method, **EmailAttachmment.setContents(byte[]),** is not modified by the eWay. User data must be encoded in **iso-2022-jp** in the byte array before the attachment content is set.

There is a problem with the Java mail interface used by the eWay whereby double byte character strings of a single character cause a java.nio exception. To get around this exception, the eWay adds a double byte space to any single character DB string where necessary (for example, in the Name field of the e-mail address).

5.6.1 Migrating ICAN 5.0 Projects that Use Japanese Encoding

The **e-Mail eWay version 5.1**, has changed the way in which Japanese Character encoding is configured. For version 5.0, Japanese encoding (iso-2022-jp) was set in the Java Collaboration. When your e-Mail eWay 5.0 Project is migrated to the Sun Java

Composite Application Platform Suite, version 5.1, the Projects Japanese encoding must be configured as described above (see **"Enabling Japanese Character Support" on page 78**), and the Business Rule or Java code previously used to enable Japanese encoding must be removed from the Collaboration. If this code is not removed, the Project will throw an exception.

Deleting the Japanese Character Encoding Business Rule

The e-Mail eWay, version 5.0, enabled Japanese encoding (iso-2022-jp) from the Java Collaboration using the following Business Rule (in the Business Rule tree:

For outbound Collaborations:

Copy "iso-2022-jp" to EmailClient_1.EmailMessage.CharSet

This adds the following Java code to the outbound Collaboration:

```
EmailClient_1.getEmailMessage().setCharSet( "iso-2022-jp" );
```

For Inbound Collaborations:

Copy "iso-2022-jp" to input.EmailMessage.CharSet

This adds the following Java code to the "Receive" Collaboration:

```
input.getEmailMessage().setCharSet( "iso-2022-jp" );
```

To remove this code, do the following:

1. Check-out your Collaboration by right-clicking the Collaboration on the Project Explorer tree and selecting **Version Control > Check Out** on the shortcut menu.
2. Double-click your Collaboration. The Collaboration opens in the Collaboration Editor.
3. From the Business Rules tree, right-click the Business Rule, as listed above, for your Collaboration, and select **Delete** on the shortcut menu.
4. From the Collaboration Editor toolbar, click the Validate icon to check your Collaboration for errors.
5. Save your current changes to the repository, and check the Collaboration back in to your Project.

Chapter 6

Using SSL

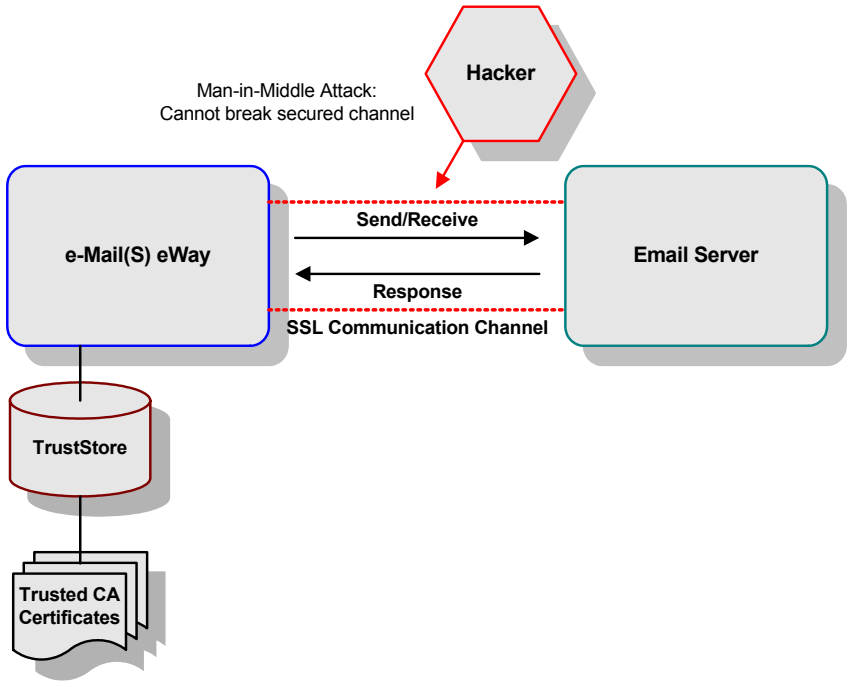
This chapter describes the operation of the eWay's Secure Sockets Layer (SSL) feature.

What's in This Chapter

6.1 Overview

The use of SSL with the e-Mail eWay enables secure e-mail exchanges, safe from unauthorized interception by "hackers" or other entities. The eWay's SSL feature provides a secure communications channel for the data exchanges (see Figure 43).

Figure 43 General SSL Operation



This SSL feature is supported through the use of JSSE version 1.0.3.

Note: *JSSE 1.0.3 if using SDK 1.3.1, or JSEE version bundled with SDK 1.4.1 release.*

Currently, the JSSE reference implementation is used. JSSE is a provider-based architecture, meaning that it implements a set of standard interfaces for cryptographic algorithms, hashing algorithms, secured-socket-layered URL stream handlers, and so forth.

Because the user is interacting with JSSE through these interfaces, the different components can be mixed and matched as long as the implementation is programmed under the published interfaces. However, some implementations may not support a particular algorithm.

The JSSE 1.0.3 application programming interface (API) is capable of supporting **SSL** versions 2.0 and 3.0 and **Transport Layer Security** (TLS) version 1.0. These security protocols encapsulate a normal bidirectional stream socket and the JSSE 1.0.3 API adds transparent support for authentication, encryption, and integrity protection. The JSSE reference implementation implements SSL version 3.0 and TLS 1.0.

For more information see the Sun Microsystem's JSSE documentation, available from the Sun Java Web site:

http://java.sun.com

6.2 TrustStores

Creating a TrustStore

For demonstration purposes, suppose you have the following CAs (CA is short for Certification Authority) that you trust: **firstCA.cert, secondCA.cert, thirdCA.cert**, located in the directory **C:\cascerts**. You can create a new TrustStore consisting of these three trusted certificates.

To create a new TrustStore use the following command:

```
keytool -import -file C:\cascerts\firstCA.cert -alias firstCA
        -keystore myTrustStore
```

You must enter this command two more times, but for the second and third entries, substitute **secondCA** and **thirdCA** for **firstCA**. Each of these command entries have the following purposes:

1. The first entry creates a KeyStore file name **myTrustStore** in the current working directory and imports the **firstCA** certificate into the TrustStore with an alias of **firstCA**. The format of **myTrustStore** is JKS.
2. For the second entry, substitute **secondCA** to import the **secondCA** certificate into the TrustStore, **myTrustStore**.
3. For the third entry, substitute **thirdCA** to import the **thirdCA** certificate into the TrustStore.

Once completed, myTrustStore is available to be used as the TrustStore for the eWay.

Using an Existing TrustStore

This section explains how to use an existing TrustStore (such as trustedcacertsjks created earlier). Notice that in the previous section, steps 2 and 3 were used to import two CAs into the TrustStore created in step 1.

For example, suppose you have a trusted certificate file named: **C:\trustedcerts\foo.cert** and want to import it to the **trustedcacertsjks** TrustStore.

If you are importing certificates into an existing TrustStore, use the following:

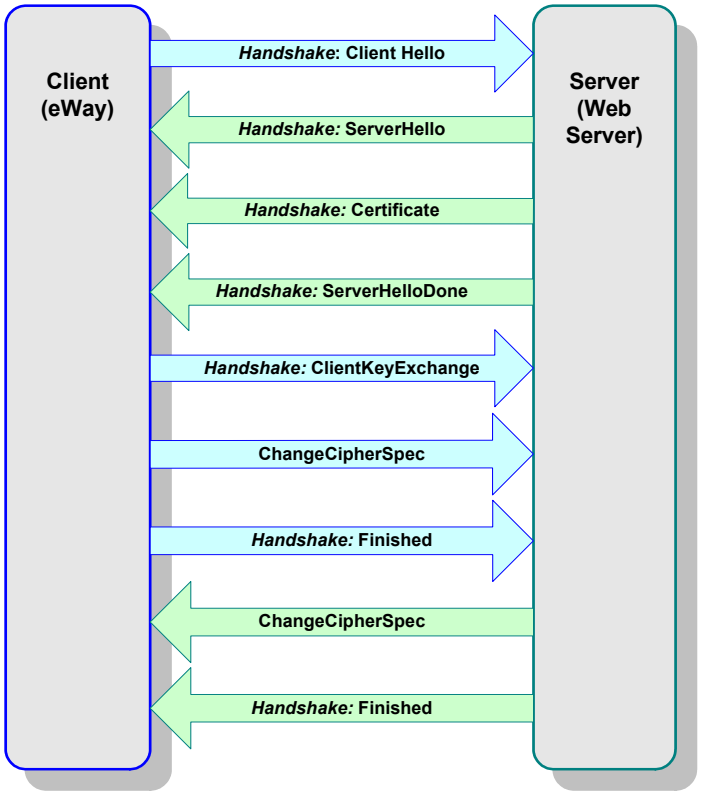
```
keytool -import -file C:\cacerts\secondCA.cert -alias secondCA
    -keystore trustedcacertsjks
```

Once you are finished, **trustedcacertsjks** can be used as the TrustStore for the eWay.

6.3 SSL Handshaking

The majority of eCommerce Web sites on the Internet are configured for **Server-Side Authentication**. The eWay requests a certificate from the Web server and authenticates the Web server by verifying that the certificate can be trusted. Essentially, the eWay performs this operation by looking into its TrustStore for a CA certificate with a public key that can validate the signature on the certificate received from the Web server. This option is illustrated in Figure 44.

Figure 44 Server-side Authentication



Index

S

T