SeeBeyond ICAN Suite

e-Mail eWay Intelligent Adapter User's Guide

Release 5.0.4

SeeBeyond®

Contents

Chapter 5

Implementing an e-Mail eWay Project 47

Chapter 6

Using SSL 69

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

- **About the e-Mail eWay Intelligent Adapter** on page 7
- **Japanese e-Mail Text Support** on page 8
- **About This Document** on page 8
- **SeeBeyond Web Site** on page 10

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 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).

In order for the e-mail to be in encoded in ISO-2022-JP (Japanese characters), the value for CharSet must be set as ISO-2022-JP (see **"Enabling Japanese Character Support" on page 66**).

1.2 About This Document

This section provides a brief outline of this user's guide.

1.2.1. Organization of Information

This book 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 ICAN Suite's eInsight Business Process Manager and the Web Services interface.

- **Chapter 5 "Implementing an e-Mail eWay Project"** 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.

1.2.2 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 Enterprise Manager. To access the full Javadoc, extract the Javadoc to an easily accessible folder, and double-click the **index.html** file.

1.2.3. 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.

1.2.4. Intended Audience

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

1.2.5. Document Conventions

The following conventions are observed throughout this document.

Table 1   Document Conventions

Text	Convention	Example
Names of buttons, files, icons, parameters, variables, methods, menus, and objects	**Bold** text	- Click **OK** to save and close. - From the **File** menu, select **Exit**. - Select the **logicalhost.exe** file. - Enter the **timeout** value. - Use the **getClassName()** method. - Configure the **Inbound** File eWay.
Command line arguments, code samples	`Fixed` font. Variables are shown in ***bold italic***.	`bootstrap -p` ***password***
Hypertext links	Blue text	See **Document Conventions** on page 9

Table 1   Document Conventions (Continued)

Text	Convention	Example
Hypertext links for Web addresses (URLs) or email addresses	<u>Blue underlined</u> text	<u>http://www.seebeyond.com</u> <u>docfeedback@seebeyond.com</u>

1.3   SeeBeyond Web Site

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

<u>http://www.seebeyond.com</u>

1.4   SeeBeyond Documentation Feedback

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

<u>docfeedback@seebeyond.com</u>

Installing the e-Mail eWay

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

What's in This Chapter

- **Supported Operating Systems** on page 11
- **System Requirements** on page 11
- **Installing the e-Mail eWay** on page 12

2.1 Supported Operating Systems

The e-Mail eWay is available on the following operating systems:

- Windows 2000, Windows XP, and Windows Server 2003
- HP Tru64 V5.1A
- HP-UX 11.0, 11i (PA-RISC), and 11i V2.0 (11.23)
- IBM AIX 5.1L and 5.2
- Red Hat Enterprise Linux AS 2.1 (Intel x86)
- Red Hat Linux 8 (Intel x86)
- Sun Solaris 8 and 9
- SuSE Linux Enterprise Server 8 (Intel x86)
- Japanese Windows 2000, Windows XP, and Windows Server 2003
- Japanese HP-UX 11.0, 11i (PA-RISC), and 11i V2.0 (11.23)
- Japanese IBM AIX 5.1L and 5.2
- Japanese Sun Solaris 8 and 9

2.2 System Requirements

The system requirements for the e-Mail eWay are the same as for eGate Integrator. For information, refer to the *SeeBeyond ICAN Suite Installation Guide*. It is also helpful to

review the **ICAN Suite Readme.txt** for any additional requirements prior to installation. The **Readme.txt** is located on the installation CD-ROM.

In addition

- To enable Web Services, you must install and configure the SeeBeyond eInsight Business Process Manager.

- The e-Mail eWay's SSL functionality is only available when using an SSL enabled e-mail server.

- For additional system requirements, see the **e-Mail eWay Readme**, accessed from the Enterprise Manager's **Documentation** tab.

- For more information about the requirements for Japanese character support, see **"Enabling Japanese Character Support" on page 66**.

2.3   Installing the e-Mail eWay

During the ICAN Suite installation process, the Enterprise Manager, a web-based application, is used to select and upload eWay and Add-on files (.sar files) from the ICAN installation CD-ROM to the Repository.

When the Repository is running on a UNIX operating system, the eWays are loaded using the Enterprise Manager on a Windows computer connected to the Repository server using Internet Explorer.

Installing the e-Mail eWay on an eGate supported system

The ICAN Suite installation process includes the following operations:

- Install the eGate Repository
- Upload products to the Repository
- Download components (including the eGate Enterprise Designer and Logical Host)

Follow the directions for installing the ICAN Suite in the *SeeBeyond ICAN Suite Installation Guide.* After you have installed eGate and any other purchased core products, do the following:

1 From the Enterprise Manager's **ADMIN** tab, browse to the **Add-ons** directory and select the **ProductsManifest.xml**, and click **Submit**. The available Add-on product list is now displayed.

2 Browse to and select the following files located in the **Add-ons** directory:

- **FileeWay.sar** (to install the File eWay, used in the sample Project)

- **eMaileWay.sar** (to install the e-Mail eWay)

3 Click **upload now** to upload the selected products.

4 Click on the Manifest File field's **Browse** option, browse to the Add-ons > **Documentation** directory, select the **ProductsManifest.xml**, and click **Submit**. The available Add-on documentation list is now displayed.

5   Select and upload the following file:

  ◆ **eMaileWayDocs.sar** (to upload the e-Mail eWay User's Guide, Javadoc, Readme, and sample Projects to the Enterprise Manager).

6   Click **upload now** to upload the selected product.

7   Continue installing the eGate Integrator as instructed in the *SeeBeyond ICAN Suite Installation Guide.*

Adding the eWay to an Existing ICAN Suite Installation

If you are installing the eWay to an existing ICAN installation, do the following:

1   Complete steps 1 through 6 above.

2   Open the Enterprise Designer and select **Update Center** from the Tools menu. The Update Center Wizard appears.

3   For Step 1 of the wizard, simply click **Next**.

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

7   When prompted, restart the IDE to complete the installation.

After Installation

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

The e-Mail eWay User Guide, Javadoc, Readme, and sample Projects, can be accessed from the Enterprise Manager's Documentation tab.

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 configuration properties with parameters that are unique to that eWay type. After an eWay is created and an External System for that eWay type is created in the Project's Environment, the eWay's properties can be modified for your specific system. The e-Mail eWay properties are modified from two locations:

- From the **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.

- From the **Environment Explorer tree**. 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 eMail External System box.

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 on page 15**). The selected e-Mail External Application icon appears on the Connectivity Map toolbar.

Figure 1   External Applications Selection Menu
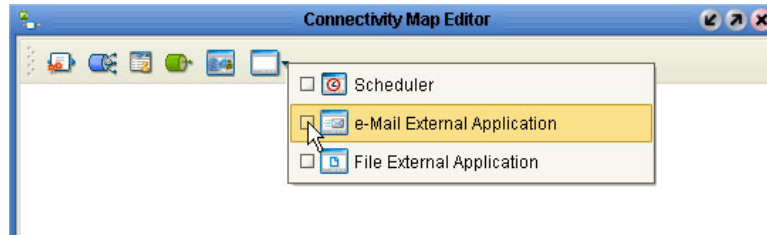
3 Drag the new **e-Mail External Application** from the toolbar onto the Connectivity Map canvas. This represents an external e-Mail system.

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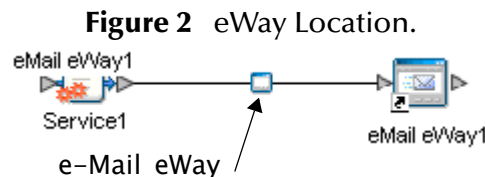
When **e-Mail** is selected as the External Application, it automatically applies the default e-Mail eWay properties, provided by the OTD, to the eWay that connects it to the Service. These properties can then be modified for your specific system using the **Properties Editor**.

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

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 Environment parameters of the e-Mail eWays, 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**). In addition, the e-Mail eWay properties will differ depending on whether the eWay is **inbound** or **outbound**. Open the Properties Editor to the e-Mail eWay's outbound Environment Explorer default properties.

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** subdirectory displays the editable parameters in the right pane, as shown in Figure 3.

Figure 3 Properties Editor -- e-Mail eWay outbound Environment Settings
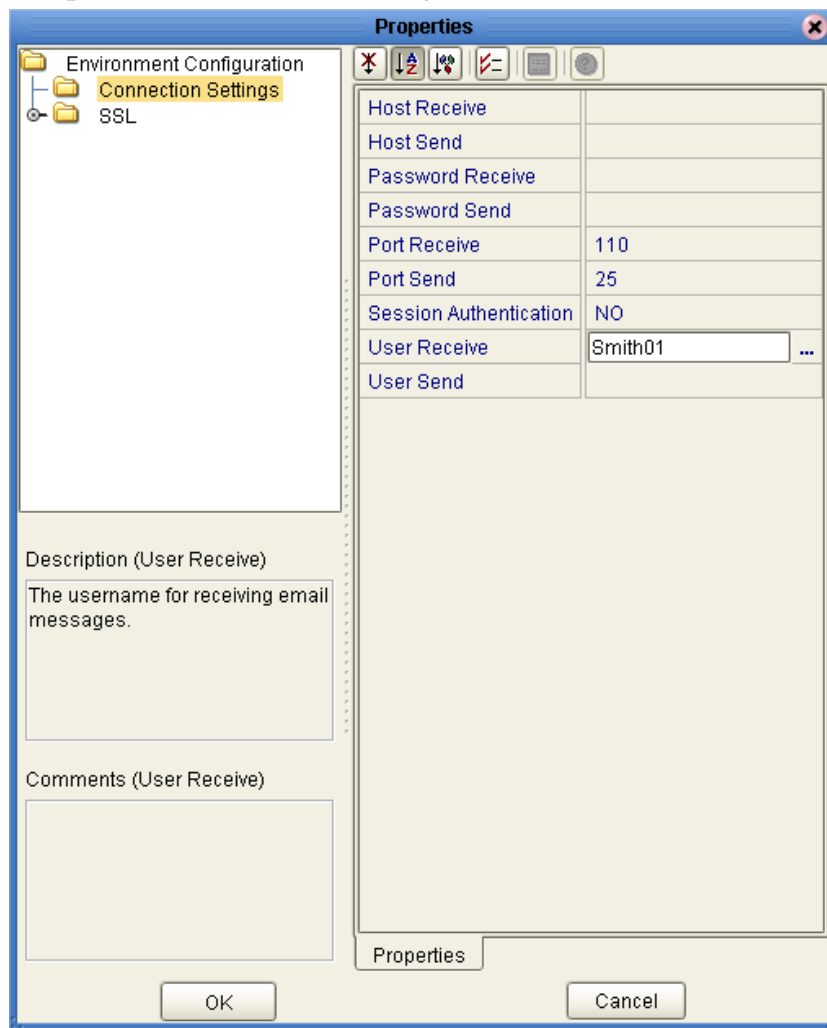
3 Click on any property field to make it editable. For example, click on the **User Receive** property to edit the user name. If a parameter's value is true/false or multiple choice, the field reveals a submenu of property options.

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 includes four different sets of parameters:

- **Inbound e-Mail eWay Connectivity Map Properties** on page 18
- **Inbound e-Mail eWay Environment Properties** on page 20
- **Outbound e-Mail eWay Connectivity Map Properties** on page 23
- **Outbound e-Mail eWay Environment Properties** on page 25

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:

▪ **SSL - Receive** on page 18

3.3.1. SSL - Receive

The **SSL - Receive** section contains the top level parameters displayed in Figure 4.

Figure 4   Properties Editor - Inbound Connectivity Map, SSL - Receive
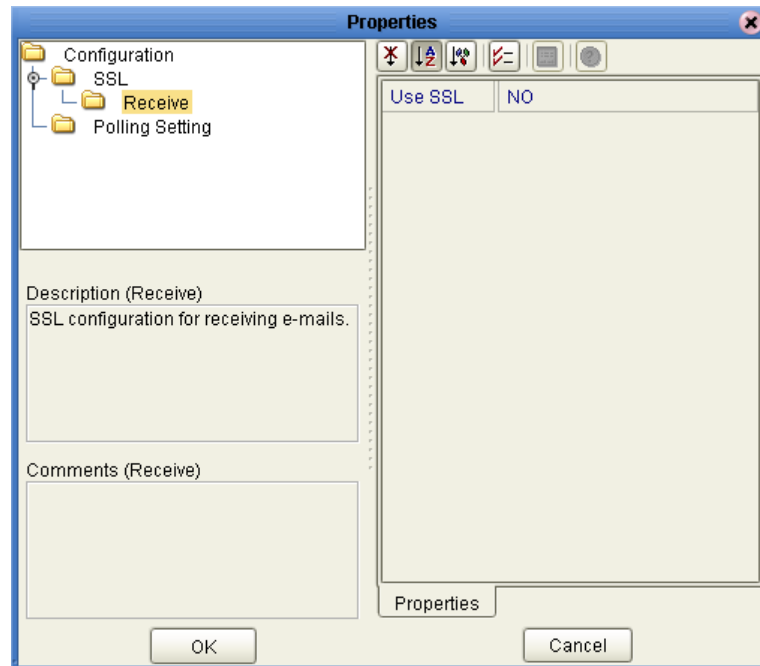
Use SSL

Description

Specifies whether SSL connections are used for accessing e-mail store.

Required Value

Yes or **No**. **Yes** indicates that SSL connections are used to access the e-mail store. The configured default is **No**.

3.3.2. Polling Setting

The **Polling Setting** section contains the top level parameters displayed in Figure 5.

Figure 5 Properties Editor - Inbound Connectivity Map, Polling Setting
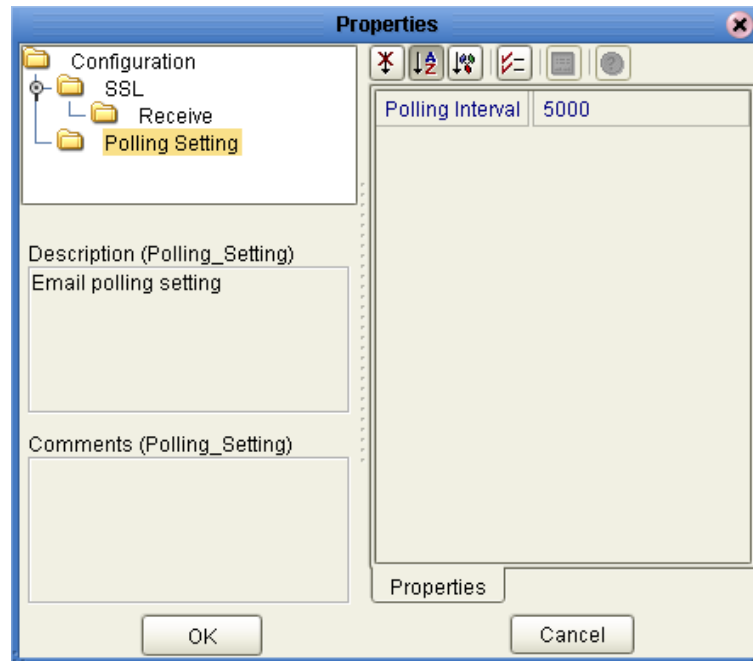
Polling Interval

Description

Specifies the interval, in milliseconds, at which the e-mail source file is polled for new incoming e-mail messages.

Required Value

A number to indicate the polling interval in milliseconds. For example 5000 milliseconds equals 5 seconds. The configured default is **5000**.

3.4   Inbound 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:

- **Connection Settings** on page 20

3.4.1. Connection Settings

The **Connection Settings** section contains the parameters displayed in Figure 6.

Figure 6   Properties Editor - Inbound Environment, Connection Settings
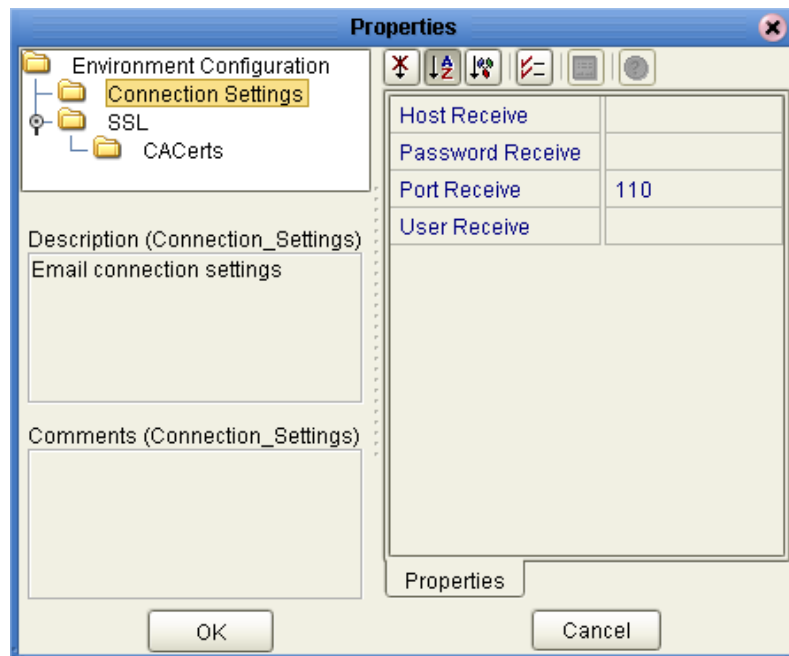
Host Receive

Description

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).

Required Value

The host name of the server used to receive messages.

Password Receive

Description

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).

Required Value

The user password used when receiving messages.

Port Receive

Description

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).

Required Value

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

Description

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).

Required Value

The valid user login name used when receiving e-mail messages.

3.4.2. **SSL - CACerts**

The **SSL - CACerts** section contains the top level parameters displayed in Figure 7.

Figure 7   Properties Editor - Inbound Environment, SSL - CACerts

TrustStore

Description

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.

Required Value

The truststore used for CA certificate management.

TrustStore password

Description

Specifies the password for accessing the truststore used for CA certificate management when establishing SSL connections.

Required Value

Integer-set. An integer in the range of 1 to 65535. The configured default is 110.

TrustStore type

Description

Specifies the truststore type.

Required Value

The trustStore type. The default TrustStore is **JKS**.

3.5 Outbound e-Mail eWay Connectivity Map Properties

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

- **SSL - Send** on page 23
- **SSL - Receive** on page 24

3.5.1. SSL - Send

The **SSL - Send** section contains the top level parameters displayed in Figure 8.

Figure 8   Properties Editor - Outbound Connectivity Map, SSL - Send
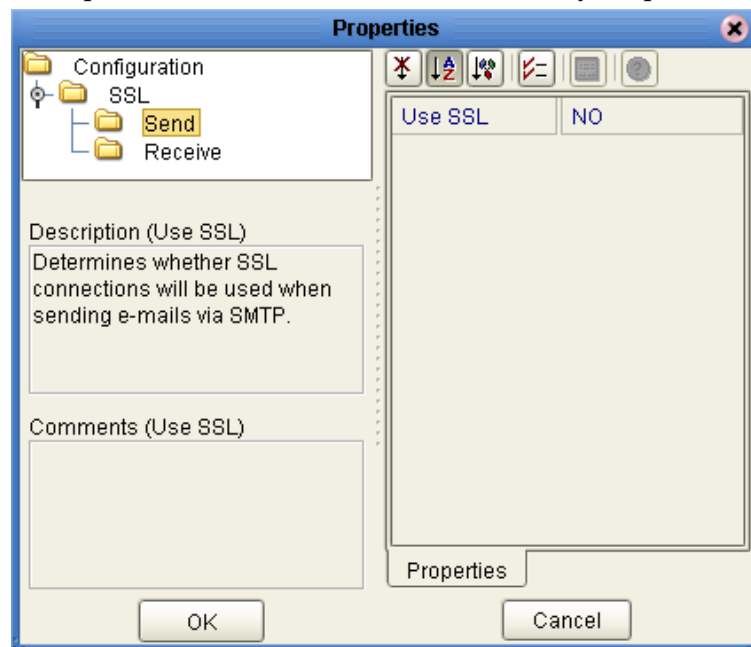
Use SSL

Description

Specifies whether SSL configuration is used when sending e-mail via SMTP.

Required Value

YES or **NO**. YES indicates that SSL configuration will be used.

3.5.2. SSL - Receive

The **SSL - Receive** section contains the top level parameters displayed in Figure 9.

Figure 9   Properties Editor - Outbound Connectivity Map, SSL - Receive
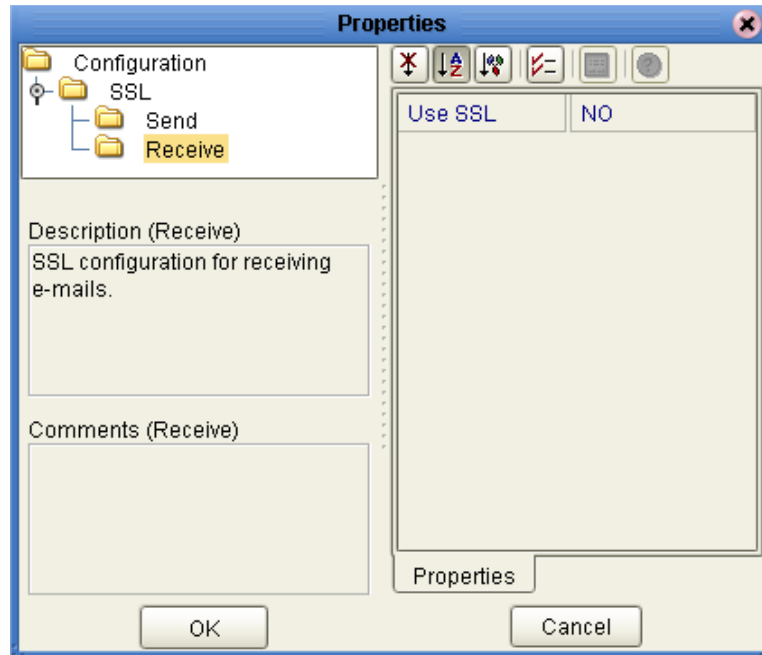
Use SSL

Description

Specifies whether the SSL configuration is used for receiving e-mail.

Required Value

YES or **NO**. YES indicates that SSL configuration will be used.

3.6 Outbound 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:

- **Connection Settings** on page 25
- **SSL - CACerts** on page 28

3.6.1. Connection Settings

The **Connection Settings** section contains the parameters displayed in Figure 10.

Figure 10 Properties Editor - Outbound Environment, Connection Settings
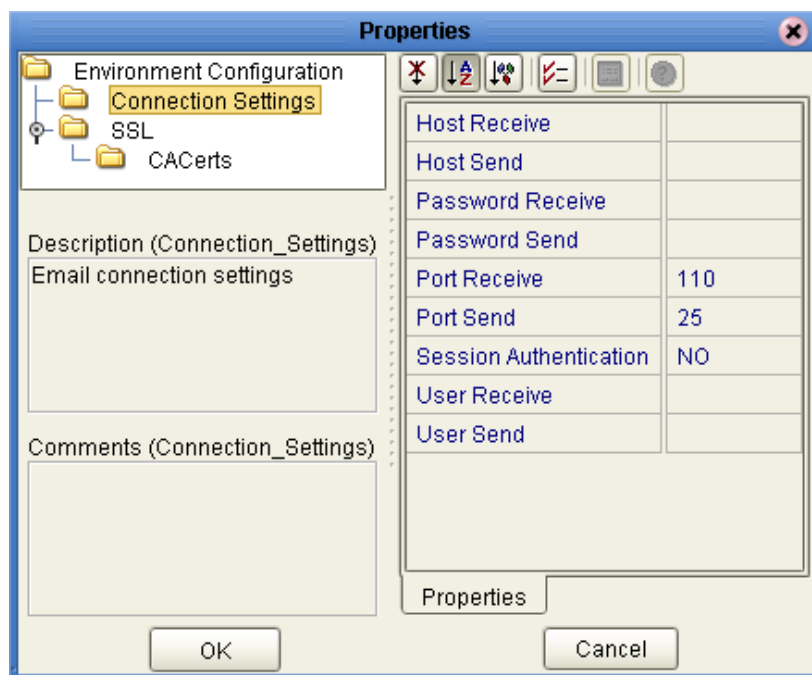
Host Receive

Description

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).
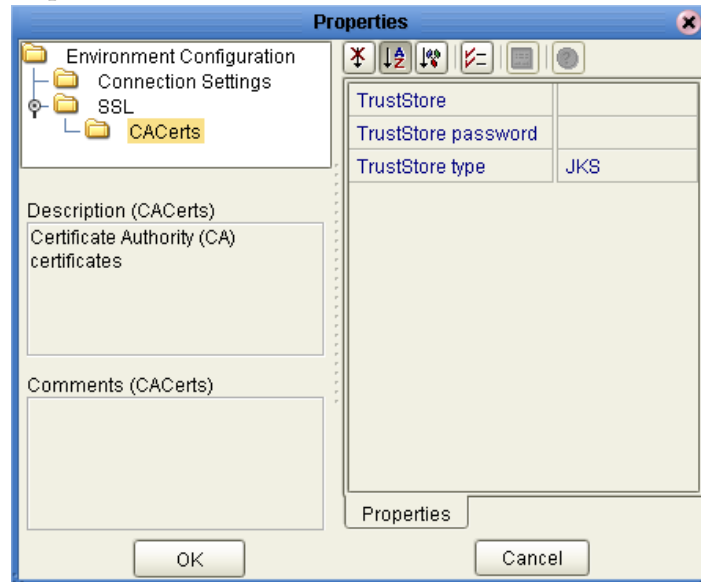
Required Value

The host name of the server used to receive messages.

Host Send

Description

Specifies the host name of the server used to send messages. This is required for "sending" eWay connection.

Required Value

The host name of the server used to send messages.

Password Receive

Description

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).

Required Value

The user login password used to access the receiving host server.

Password Send

Description

Specifies the password used when sending messages. This is required for "sending" eWay connections.

Required Value

The user login password used to access the sending host server.

Port Receive

Description

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).

Required Value

The port number used to connect with the receiving host server. This is an integer in the range of 1 to 65535. The configured default is **110**.

Port Send

Description

Specifies the port number to connect to when sending messages. This is required for "sending" eWay connections.

Required Value

The port number used to connect with the sending host server. This is an integer in the range of 1 to 65535. The configured default is **25**.

Session Authentication

Description

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.

Required Value

YES or **NO**. YES indicates that POP3 session authentication will be performed before attempting an SMTP connection.

User Receive

Description

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).

Required Value

The user login name used to access the receiving host server.

User Send

Description

Specifies the user name used when sending messages. This is required for "sending" eWay connections.

Required Value

The user login name used to access the sending host server.

3.6.2. **SSL - CACerts**

The **SSL - CACerts** section contains the top level parameters displayed in Figure 11.

Figure 11   Properties Editor - Outbound Environment, SSL - CACerts

TrustStore

Description

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.

Required Value

The truststore used for CA certificate management.

TrustStore password

Description

Specifies the password for accessing the truststore used for CA certificate management when establishing SSL connections.

Required Value

Integer-set. An integer in the range of 1 to 65535. The configured default is 110.

TrustStore type

Description

Specifies the truststore type.

Required Value

The trustStore type. The default TrustStore is **JKS**.

Using the e-Mail eWay With eInsight

This chapter describes how to use the e-Mail eWay with the ICAN 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

- **The eInsight Engine and Components** on page 29
- **The e-Mail eWay With eInsight** on page 30
- **Importing a Sample Project** on page 31
- **The eMailSampleBPEL Project Overview** on page 31
- **Alerting and Logging** on page 46

4.1 The eInsight Engine and Components

eGate components can be deployed as Activities in eInsight Business Processes. Once a component is associated with an Activity, eInsight invokes it using a Web Services interface. eGate components that can interface with eInsight in this way include the following:

- Object Type Definitions (OTDs)
- eWays
- Collaborations

Using the Enterprise Designer and eInsight, you can add an Activity to a Business Process, then associate that Activity with an eGate component, for example, an eWay. Then, when eInsight runs the Business Process, it automatically invokes that component via its Web Services interface.

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.

- 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.

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 can be uploaded from the Installation CD-ROM in a near-complete state.

▪ For more information on creating ICAN Projects see the *eInsight Business Process Manager User's Guide* and the *eGate Integrator User's Guide*.

4.5 Importing a Sample Project

Sample eWay Projects are included as part of the installation CD-ROM 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 Enterprise Manager's Documentation tab. The **eMail_eWay_Sample.zip** file contains the various sample Project zip files. Extract the samples from the Enterprise Manager to a local file.

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

3 Browse to the directory that contains the sample Project zip file. Select the sample file (for this sample, **eMailSampleBPEL.zip**) and click **Import**. After the sample Project is successfully imported, click **Close**.

4 When the import has successfully completed, right-click the Repository and select **Refresh All from Repository** from the shortcut menu.

5 Before an imported sample Project can be run you must do the following:

 ◆ Create an **Environment**

 ◆ Configure the eWays for your specific system

 ◆ Create a **Deployment Profile**

4.6 The eMailSampleBPEL Project Overview

The e-Mail eWay Project, **eMailSampleBPEL**, 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 and publishes the message to the bp_Send Business Process.

▪ The bp_Send Business Process adds a "To" name and e-mail address, and a "From" name and e-mail address, the subject name, and the text message and publishes the message to the outbound e-Mail eWay.

- The outbound e-Mail eWay publishes the e-mail message to a POP3 server.

Receive Message

- The inbound e-Mail eWay subscribes to an external e-mail account, polling the account every 60 seconds. Email is published to the bp_Receive Business Process.

- The bp_Receive Business Process parses the sender address, writes it to a text file and publishes the text file to the outbound File eWay.

- The outbound File eWay receives the text file and publishes the file to an external directory.

4.7 Creating the eMailSampleBPEL Project

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

4.7.1. Creating a Project

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

1 Start the Enterprise Designer.

2 From the Project Explorer tree, right-click the Repository and select **New Project** (see Figure 12). A new Project (**Project1**) appears on the Project Explorer tree.

Figure 12   Enterprise Explorer - New Project
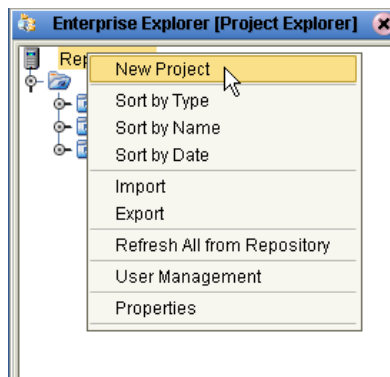
3 Click twice on **Project1** and rename the Project (for this sample, **eMailSampleBPEL**).

4.7.2 Creating a Business Process

The **eMailSampleBPEL** contains two Business Processes:

- **bp_Send**: Used to send e-mail messages

- **bp_Receive**: Used to receive e-mail messages

To create the e-Mail eInsight Business Process, do the following:

Creating the bp_Send Business Process

1   Right-click the Project (for this sample **eMailSampleBPEL**) in the Enterprise Designer's Project Explorer, and select **New** > **Business Process** from the shortcut menu. The eInsight Business Process Designer appears and **BusinessProcess1** is added to the Project Explorer tree. Rename **BusinessProcess1** to **bp_Send**.

2   From the Project Explorer tree, expand the 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 13 on page 33**:

   ◆ **receive**, under SeeBeyond > eWays > File > FileClient

   ◆ **sendMessage**, under SeeBeyond > eWays > EmaileWay > EmailClient

Figure 13   eInsight Business Process Designer - Populate the Canvas
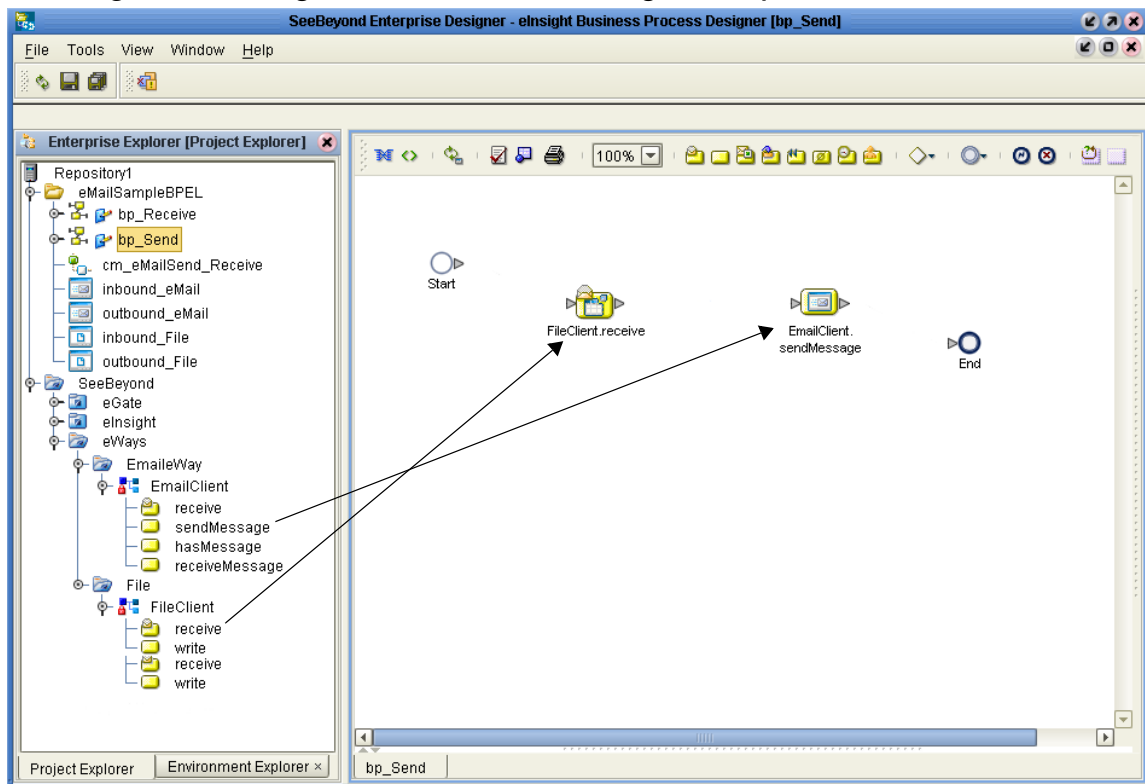
4   Link the modeling elements by clicking on the element connector and dragging the cursor to the next element connector, making the following links as displayed in Figure 14.

   ▪ **Start** -> **FileClient.receive**

   ▪ **FileClient.receive** -> **EmailClient.sendMessage**

   ▪ **EmailClient.sendMessage** -> **End**

Figure 14 Business Process Designer - Link the Modeling Elements
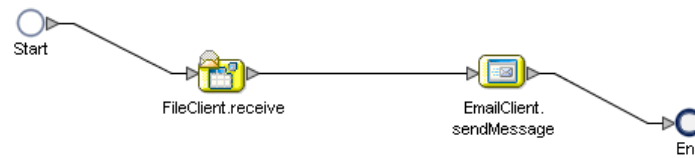
Configuring the bp_Send Modeling Elements

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. To create the **bp_Send** Business Rules do the following:

Adding Business Rules

  1 Right-click the link between the **FileClient.receive** and **EmailClient.sendMessage** Activities and select **Add Business Rule** from the shortcut menu (see Figure 15).

Figure 15 eInsight Business Process Designer - Adding Business Rules

Using the Business Rule Designer

  1 From the eInsight Business Process Designer toolbar, click the **Map Business Process Attributes** button. The Business Rule Designer appears at the bottom of the eInsight Business Process Designer.

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

  3 Map **text**, under **FileClient.receive.Output** in the Output pane of the Business Rule Designer, to **msgText** under **EmailClient.sendMessage.Input** > **emailMessage** > **replyTo** in the Input pane of the Business Rule Designer. To do this, click on **text** in the Output pane and drag the cursor to **msgText** in the Input pane. A line now connects the two nodes in the Business Rule Designer (see **Figure 17 on page 37**).

  4 From the Business Rule Designer toolbar, click **string-literal** in the String drop-down menu. The **Input** dialog box appears. Enter **maileway** as the literal value and click **OK**. Map the **maileway** output node to **name** under **EmailClient.sendMessage.Input** > **e-mailMessage** > **to[1]** in the Input pane of the Business Rule Designer (see **Figure 17 on page 37**).

5 Again, click **string-literal** in the String drop-down menu. The **Input** dialog box appears. Enter your sample input address as the literal value and click **OK**. Map the string literal's selected address output node to **address** under **EmailClient.sendMessage.Input** > **emailMessage** > **to[1]** in the Input pane of the Business Rule Designer (see **Figure 17 on page 37**).

6 Again, click **string-literal** in the String drop-down menu. Enter **maileway** as the literal value and click **OK**. Map the string literal's **maileway** output node to **address** under **EmailClient.sendMessage.Input** > **emailMessage** > **from** in the Input pane of the Business Rule Designer.

7 Again, click **string-literal** in the String drop-down menu, enter your "from" address as the literal value and click **OK**. Map the string literal's selected address output node to **address** under **EmailClient.sendMessage.Input** > **emailMessage** > **from** in the Input pane of the Business Rule Designer.

8 Once more, click **string-literal** in the String drop-down menu, enter **eMail eWay** as the literal value and click **OK**. Map the string literal's **eMail eWay** output node to **address** under **EmailClient.sendMessage.Input** > **emailMessage** > **from** in the Input pane of the Business Rule Designer.

Figure 16   Business Rule Designer - bp_Send Business Process
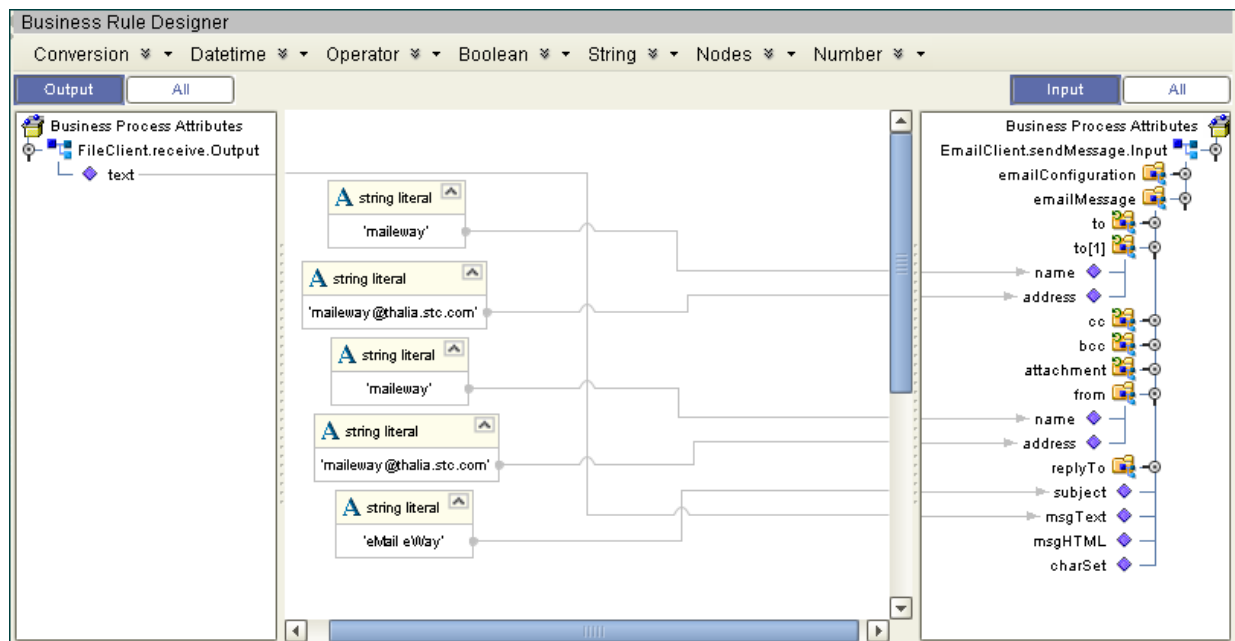
9 From the Business Process Designer toolbar, click the **Synchronize Graphical Model and Business Process** icon to synchronize the graphical interface to the Business Process code.

10 Save your changes to the Repository.

Creating the bp_Receive 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 **bp_Receive**.

2 Populate the eInsight Business Process Designer's modeling canvas with the following activities from the Project Explorer tree:

   ◆ **receive**, under SeeBeyond > eWays > EmaileWay > EmailClient

   ◆ **write**, under 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.

▪ **Start** -> **EmailClient.receive**

▪ **EmailClient.receive** -> **FileClient.write**

▪ **FileClient.write** -> **End**

Configuring the bp_Receive Modeling Elements

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

Adding Business Rules

1 Right-click the link between the **EmailClient.receive** and **FileClient.write** Activities and select **Add Business Rule** from the shortcut menu.

Using the Business Rule Designer

1 From the eInsight Business Process Designer toolbar, click the **Map Business Process Attributes** button. The Business Rule Designer appears at the bottom of the eInsight Business Process Designer.

2 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.

3 Map **name**, under **EmailClient.receive.Output** > **emailMessage** > **from** in the Output pane of the Business Rule Designer, to **text** under **FileClient.write.Input** in the Input pane of the Business Rule Designer. To do this, click on **name** in the Output pane and drag the cursor to **text** in the Input pane. A line now connects the two nodes in the Business Rule Designer (see Figure 17).

Figure 17   Business Rule Designer - bp_Receive Business Process
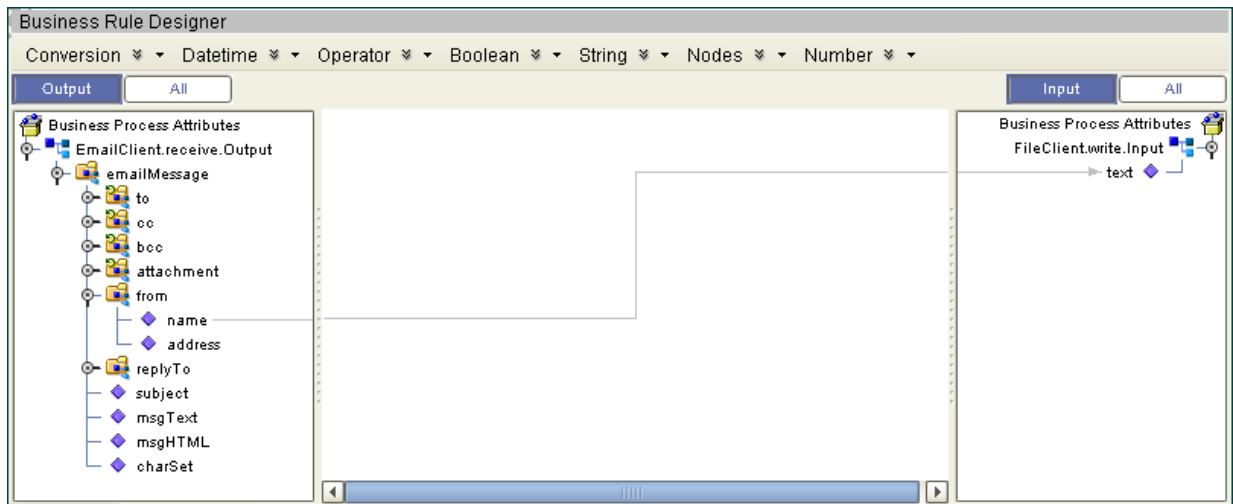
4 From the Business Process Designer toolbar, click the **Synchronize Graphical Model and Business Process** icon to synchronize the graphical interface to the Business Process code.

5 Save your changes to the Repository.

4.7.3 Creating a Connectivity Map

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

1 From the Project Explorer tree, right-click the new **eMailSampleBPEL** 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 **cm_eMailSend_Receive**.

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

Selecting the External Applications

In the Connectivity Map, the 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 18).

Figure 18 Connectivity Map - External Applications
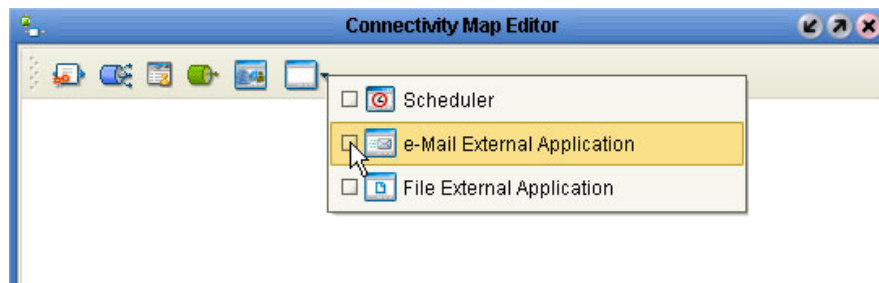
1 Click the **External Application** icon on the Connectivity Map toolbar,

2 Select the external systems necessary to create your Project (for this sample, **EmaileWay** and **File**). Icons representing the selected external systems are added to the Connectivity Map toolbar.

Populating the Connectivity Map

Add the Project components to the Connectivity Map 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 19:

  ◆ **File External System** (2 for this sample).

  ◆ **Service** (2 for this sample) A service is a container for Java Collaborations, Business Processes, eTL processes, and so forth.

  ◆ **e-Mail External System** (2 for this sample)

2 Rename the **File1** External Application to **inbound_File** by right-clicking the object, selecting **Rename** from the shortcut menu, and typing in the new name.

3 Rename the **File2** External Application to **outbound_File**.

4 Rename the **EmaileWay1** External Application to **inbound_eMail**.

5 Rename the **EmaileWay2** External Application to **outbound_eMail**.

6 Rename the **Service1** Service to **bp_Send1**.

7 Rename the **Service2** Service to **bp_Receive1** (see Figure 19).

8 Save your current changes to the Repository.

Figure 19   Connectivity Map with Components
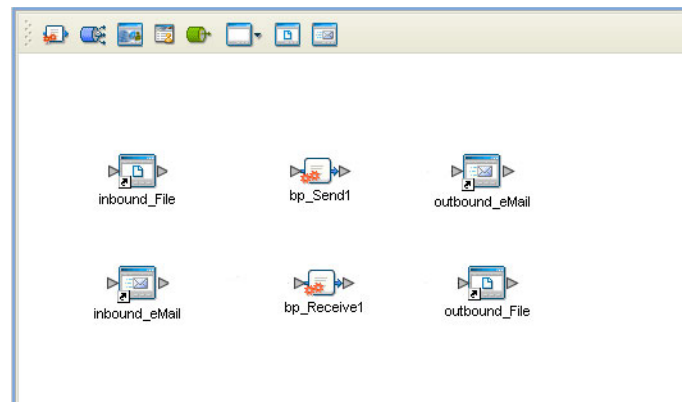
4.7.4. Binding the eWay Components

After the Business Processes have been completed, the components are associated and Bindings are created in the Connectivity Map.

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

2   Drag and drop the **bp_Send** Business Process from the Project Explorer onto the Service (**bp_Send1**). If the Business Process was successfully associated, the Service's icon changes to a Business Process icon (see Figure 20).

3   Drag and drop the **bp_Receive** Business Process from the Project Explorer onto the Service (**bp_Receive1**).

Figure 20   Connectivity Map - Binding the eWay Components
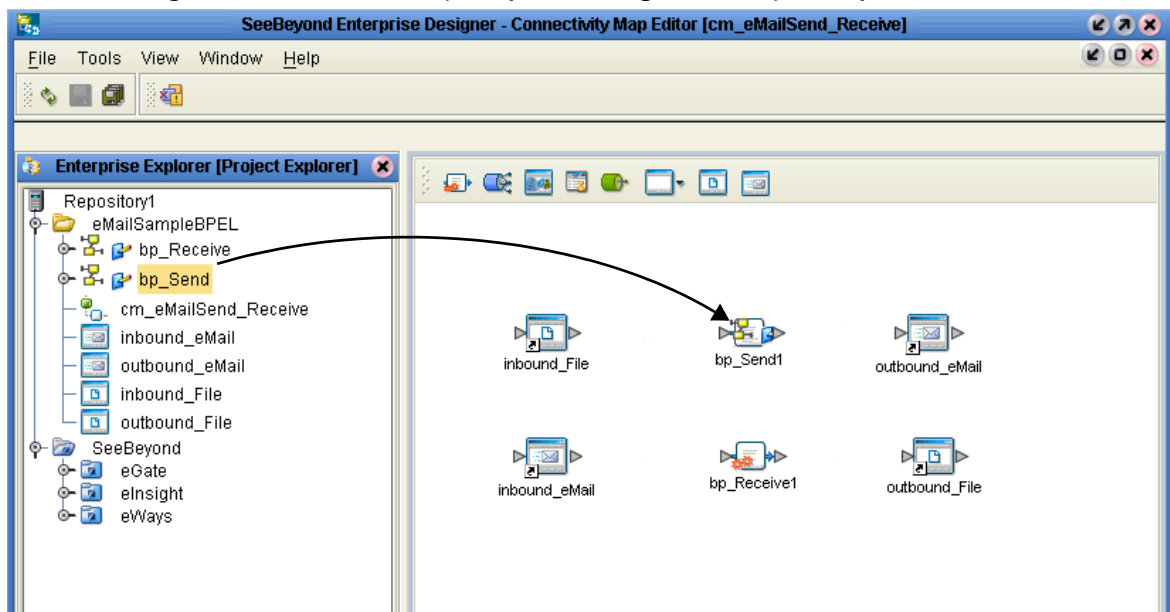
4   Double-click **bp_Send1**. The **bp_Send1** binding dialog box appears using the **bp_Send** Rule.

5   From the **bp_Send1** binding dialog box, drag **FileSender** (under Implemented Services) to the **inbound_File** (File) External Application.

6   From the **bp_Send1** binding dialog box, drag **EmaileWayReceiver** (under Invoked Services) to the **outbound_eMail** External Application (see **Figure 21 on page 40**). Minimize the **bp_Send1** binding dialog box by clicking the chevrons in the upper-right corner.

Figure 21   Connectivity Map - Associating (Binding) the Project's Components

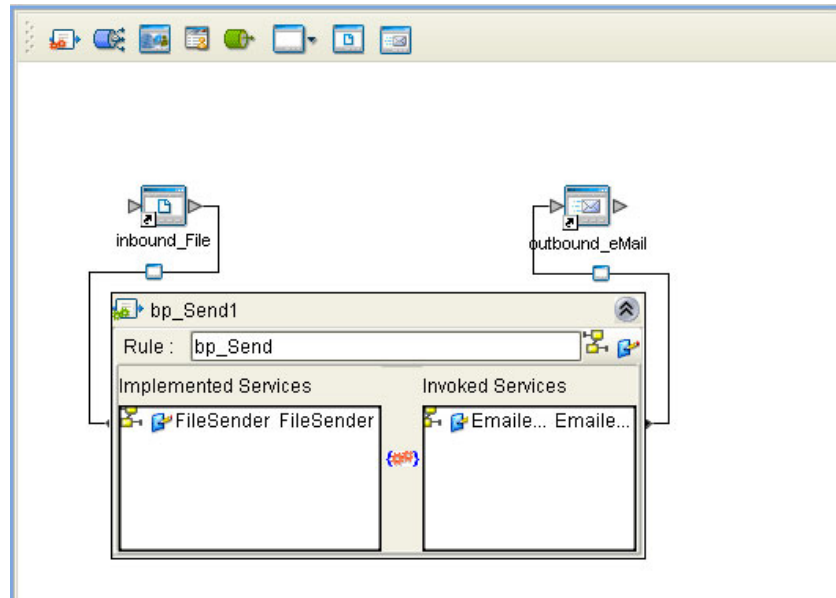7   Double-click **bp_Receive1**. The **bp_Receive1** binding dialog box appears using the **bp_Receive** Rule.

8   From the **bp_Receive1** binding dialog box, drag **EmaileWaySender** (under Implemented Services) to the **inbound_eMail** External Application.

9   From the **bp_Receive1** binding dialog box, drag **FileReceiver** (under Invoked Services) to the **outbound_File** External Application. Minimize the **bp_Receive1** binding dialog box by clicking the chevrons in the upper-right corner.

10   Minimize the **bp_Receive1** binding dialog box by clicking the chevrons in the upper-right corner.

11   Save your current changes to the Repository.

## 4.7.5. Creating an Environment

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

4   Right-click **eMailSampleBPEL_ENV** and select **New e-Mail External System**. Name the External System **EmailExtSysIn** and select Inbound Email eWay as the External System Type. Click **OK**. **EmailExtSysIn** is added to the Environment Editor.

5   Right-click **eMailSampleBPEL_ENV** and select **New e-Mail External System**. Name the External System **EmailExtSysOut** and select Outbound Email eWay as the External System Type. Click **OK**. **EmailExtSysOut** is added to the Environment Editor.

6   Right-click **eMailSampleBPEL_ENV** and select **New File External System**. Name the External System **FileExtSysIn** and select **Inbound File eWay** as the External System Type. Click **OK**. **FileExtSysIn** is added to the Environment Editor.

7   Right-click **eMailSampleBPEL_ENV** and select **New File External System**. Name this External System **FileExtSysOut** and select Outbound File eWay as the External System Type. **FileExtSysOut** is added to the Environment Editor.

8   Right-click **eMailSampleBPEL_ENV** and select **New Logical Host**. **LogicalHost1** is added to the Environment Editor.

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

10   Save changes to the repository. The Environment Explorer and Environment Editor now appear as displayed in **Figure 22 on page 41**.

Figure 22   Environment Editor
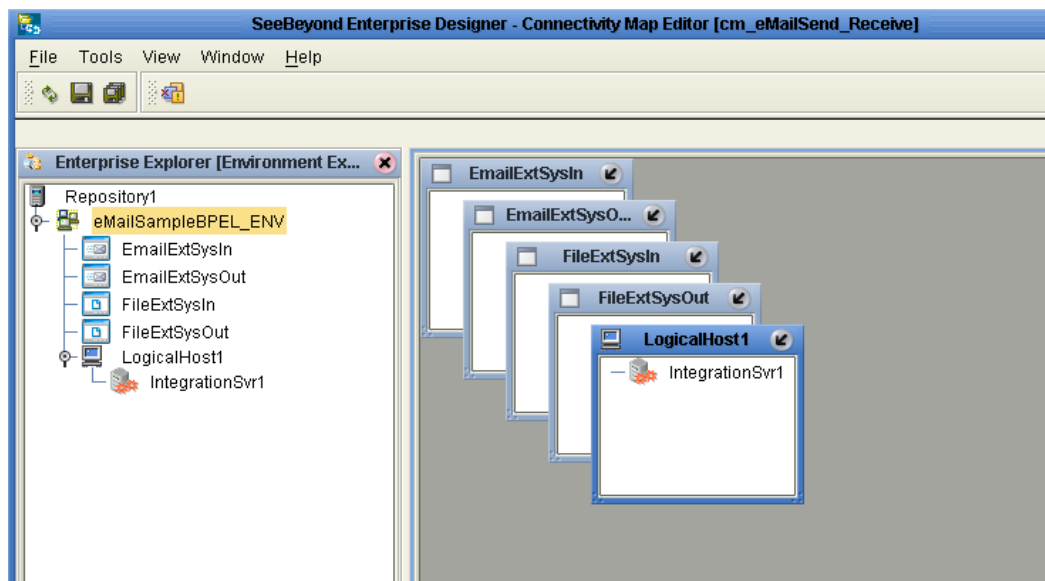
4.7.6. Configuring the eWay Properties

The eMailSampleBPEL Project uses four eWays, each represented in the Connectivity Map 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.

The File eWay properties are configured from the Connectivity Map. The e-Mail eWay properties are set from both the Project Explorer's Connectivity Map and the Environment Explorer tree.

Configuring the File eWay Properties

1 Double-click the inbound **File eWay** (see Figure 23), select **Inbound File eWay** in the Templates dialog box and click **OK**.
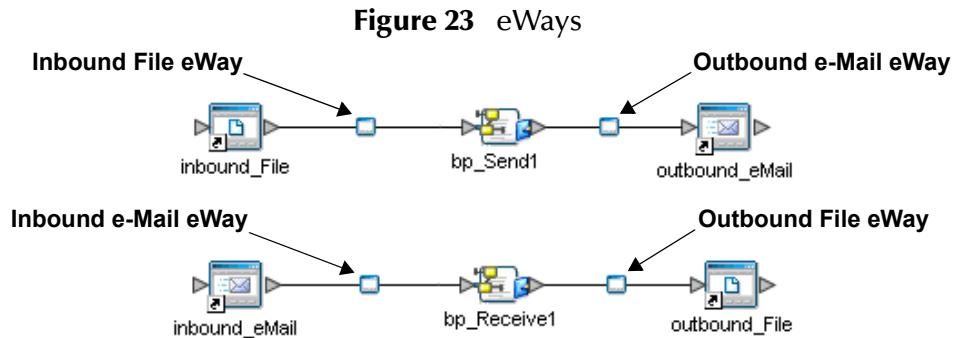
Figure 23   eWays

2 The **Properties Editor** opens to the inbound File eWay properties. Modify the properties for your system, including the settings for the inbound File eWay in Table 2, and click **OK**.

Table 2   Inbound File eWay Settings

Inbound eWay Connection Parameters	
Directory	C:/temp
Input file name	Input*.txt
Polling interval	60000

3 In the same way, modify the outbound File eWay properties for your system, including the settings in Table 3, and click **OK**.

Table 3   Outbound File eWay Settings

Outbound eWay Connection Parameters	
Directory	C:/temp

Configuring the e-Mail eWay Properties

The e-Mail eWay properties must be set in both the Connectivity Map and the Environment Explorer. For more information on the e-Mail eWay properties and the Properties Editor, see **"Configuring the e-Mail eWay Properties" on page 14** or see the *eGate Integrator User's Guide*.

Modifying the e-Mail eWay Connectivity Map Properties

1 From the **Connectivity Map**, double-click the inbound **e-Mail eWay**, select **Inbound Email eWay** in the Templates dialog box and click **OK**. The Properties Editor opens to the Inbound e-Mail eWay properties.

2 Modify the Inbound e-Mail eWay Connectivity Map properties for your system, including the settings in Table 4, and click **OK**.

Table 4   Inbound e-Mail eWay Connectivity Map Properties

Inbound e-Mail eWay (Connectivity Map) Properties	
Polling Setting Set as directed, otherwise use the default settings	
Polling Interval	60000

3 The **Outbound e-Mail eWay** Connectivity Map properties contain only SSL related parameters. Use the default for these settings.

Modifying the e-Mail eWay Environment Explorer Properties

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

2 Modify the inbound e-Mail eWay environment properties for your system, including the settings in Table 5, and click **OK**.

Table 5   Inbound e-Mail Environment Explorer eWay Properties

Inbound e-Mail eWay Environment Properties	
Connection Settings Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Password Receive	*User password for receiving e-mail*
Port Receive	*TCP/IP Port Number for receiving e-mail*
User Receive	*User name for receiving e-mail*

1 From the **Environment Explorer** tree, right-click the outbound e-Mail eWay external system (**EmailExtSysOut** in this sample), and select **Properties** from the shortcut menu. The Properties Editor appears.

2 Modify the outbound e-Mail eWay environment properties for your system, including the settings in **Table 6 on page 44**, and click **OK**.

Table 6 Outbound e-Mail Environment Explorer eWay Properties

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

4.7.7 Creating and Activating the Deployment Profile

Deployment Profiles are specific instances of a Project in a particular Environment. A Deployment Profile contains information about 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.

To create the eMailSampleBPEL Deployment Profile do the following:

1   From the Enterprise Explorer's Project Explorer, right-click the Project (**eMailSampleBPEL**) and select **New** > **Deployment Profile** from the shortcut menu.

2   Enter a name for the Deployment Profile (for this sample **eMailSampleBPEL_DP**). Make sure that the selected Environment is **eMailSampleBPEL_ENV**. Click **OK**. The Deployment Profile Editor appears.

3   Click the **Auto Map** icon as displayed in **Figure 24 on page 45**. The Project's components are automatically mapped to their system window as seen in **Figure 25 on page 45**. If any of the Project components are not mapped automatically after Auto Map is used, those component can be mapped manually by following the appropriate steps below. Once all components are mapped, proceed to step 9.

Figure 24 Deployment Profile - Auto Map
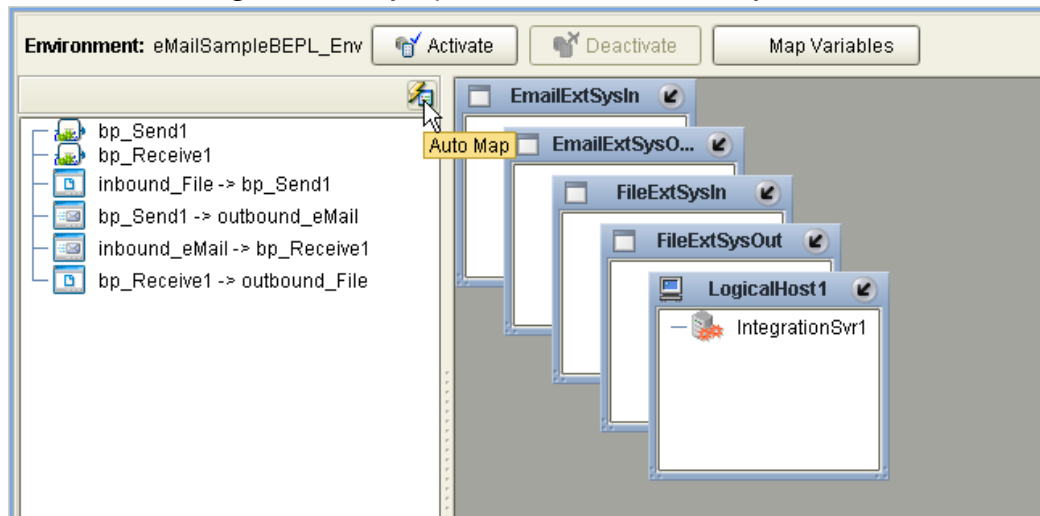
4   From the left pane of the Deployment Editor, drag the **inbound_File -> bp_Send1** (External Application) object to the **FileExtSysIn** window.

5   Drag the **bp_Receive1 -> outbound_File** (External Application) object to the **FileExtSysOut** window.

6   Drag the **bp_Send1 -> outbound_eMail** (External Application) object to the **EmailExtSysOut** window.

7   Drag the **inbound_eMail -> bp_Send1** (External Application) object to the **EmailExtSysIn** window.

8   Drag the **bp_Send1** and **bp_Receive1**(Services) to **IntegrationSvr1** in the **LogicalHost1** window (see Figure 25).

Figure 25 Deployment Profile
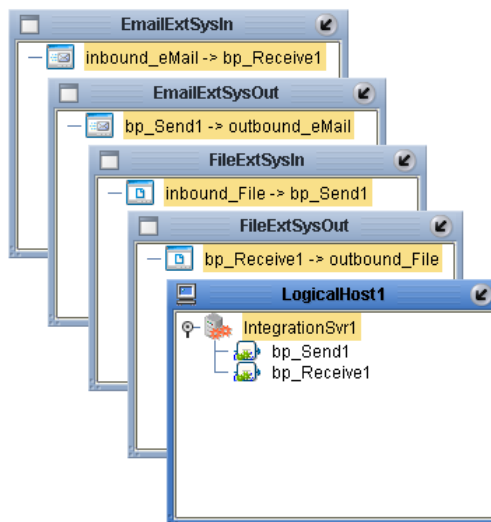
9   Click **Activate**. When activation succeeds, save your current changes to the Repository.

4.7.8. ## Running the Project

The following directions assume that the Enterprise Designer was downloaded to C:\ican50. If this is not the case, replace that location in the following directions with the appropriate location.

1   From the Enterprise Manager Downloads tab, download **Logical Host**.

2   Extract the file to the **ican50\LogicalHost1** directory. You must specify the **LogicalHost1** directory for it to be created.

3   Navigate to **C:\ican50\LogicalHost1\logicalhost\bootstrap\config** directory and open the **logical-host.properties** file using Notepad™.

4   Enter the following information in the appropriate fields:

   ◆ Logical Host root directory: **ican50\LogicalHost1\logicalhost**

   ◆ Repository URL: **http://localhost:***port number/repository name*

   ◆ Repository user name and password: *Your user name and password*

   ◆ Logical Host Environment name: **eMailSampleBPEL_Env**

   ◆ Logical Host name: **LogicalHost1**

   Save your changes to **logical-host.properties** and close the file.

5   Run the **bootstrap.bat** file in the **ican50\LogicalHost1\logicalhost\bootstrap\bin** directory.

6   Copy the sample input data file to the input directory.

4.8   # Alerting and Logging

eGate provides an alerting and logging feature. This allows monitoring of messages, and captures any adverse messages in order of severity based on configured severity level and higher. To enable Logging, please see the *eGate Integrator User's Guide*.

Note: *The alerts/status notifications for the e-Mail eWay are currently limited to Started, Running, Stopping, and Stopped.*

Implementing an e-Mail eWay Project

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 SeeBeyond Enterprise Designer. For more information on creating an eGate Project see the *eGate Tutorial* and the *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 *eGate Tutorial*. The eWay 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 14**.

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 CD-ROM 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 Enterprise Manager's Documentation tab. The **eMail_eWay_Sample.zip** file contains the various sample Project zip files. Extract the samples from the Enterprise Manager to a local file.

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

3 Browse to the directory that contains the sample Project zip file. Select the sample file (for this sample, **eMailSampleJCE.zip**) and click **Import**. After the sample Project is successfully imported, click **Close**.

4 When the import has successfully completed, right-click the Repository and select **Refresh All from Repository** from the shortcut menu.

5 Before an imported sample Project can be run you must do the following:

    - Create an **Environment**

    - Configure the eWays for your specific system

    - Create a **Deployment Profile**

5.4   The eMailSampleJCE Project Overview

The e-Mail eWay Project, **eMailSampleJCE**, 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 and publishes the message to the collabSend Collaboration.

- The collabSend Collaboration adds a "To" name and e-mail address, and a "From" name and e-mail address, the subject name, and the text message and publishes the message to the outbound e-Mail eWay.

- The outbound e-Mail eWay publishes the e-mail message to a POP3 server.

Receive Message

- The inbound e-Mail eWay subscribes to an external e-mail account, polling the account every 60 seconds. Email is published to the collabReceive Collaboration.

- The collabReceive Collaboration parses the sender address, writes it to a text file and publishes the text file to the outbound File eWay.

- The outbound File eWay receives the text file and publishes the file to an external directory.
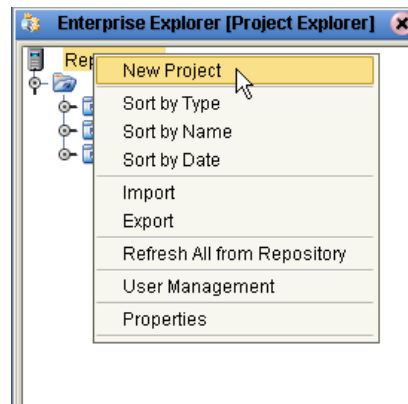
5.5   Creating the eMailSampleJCE Project

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 SeeBeyond Enterprise Designer.

1 Start the Enterprise Designer.

2 From the Project Explorer tree, right-click the Repository and select **New Project** (see **Figure 26 on page 50**). A new Project (**Project1**) appears on the Project Explorer tree.

Figure 26 Enterprise Explorer - New Project

3   Click twice on **Project1** and rename the Project (for this sample, **eMailSampleJCE**).

5.5.2 Creating a Connectivity Map

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

1   From the Project Explorer tree, right-click the new **eMailSampleJCE** 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 to **cm_eMailSend_Receive**.

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

Selecting the External Applications

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

In a Connectivity Map, the eWays are associated with External Systems. For example, to establish a connection to e-Mail, you must first select e-Mail as an External System to use in your Connectivity Map (see Figure 27).

Figure 27 Connectivity Map - External Applications
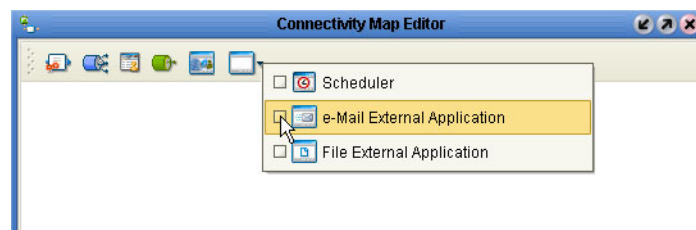
1   Click the **External Application** icon on the Connectivity Map toolbar,

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

Populating the Connectivity Map

Add the Project components to the Connectivity Map 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 28:

* **File External System** (2 for this sample)

* **Service** (2 for this sample) A service is a container for Java Collaborations, Business Processes, eTL processes, and so forth.

* **e-Mail External System** (2 for this sample)

Figure 28 Connectivity Map with Components
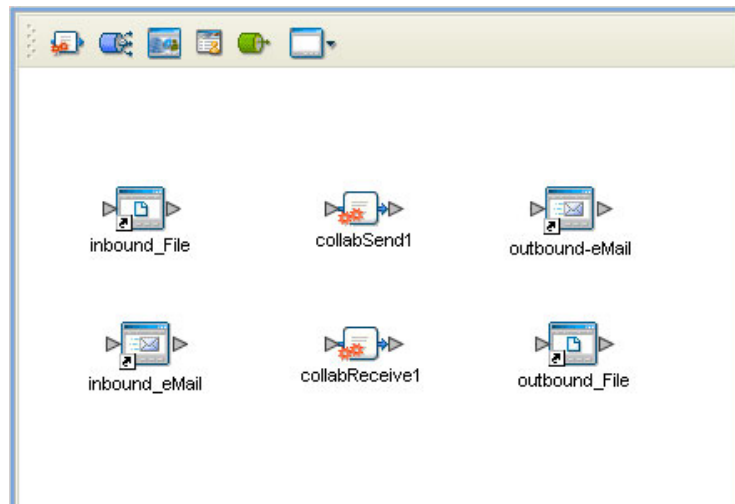
2 Rename the objects by right-clicking the object, selecting **Rename** from the shortcut menu, and typing in the new name. Change the names as follows:

* **File1** External Application to **inbound_File**

* **File2** External Application to **outbound_File**

* **EmaileWay1** External Application to **inbound_eMail**

* **EmaileWay2** External Application to **outbound_eMail**

* **Service1** Service to **collabSend1**

* **Service2** Service to **collabReceive1** (see Figure 28).

3 Save your current changes to the Repository.
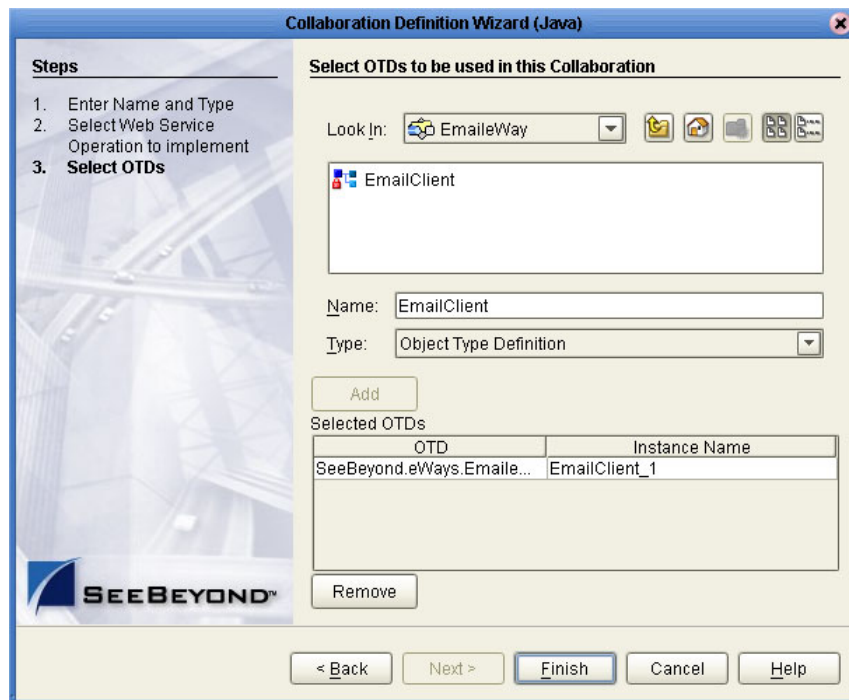
5.5.3. Creating the Collaboration Definitions

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

Creating the collabSendCollaboration Collaboration Definition

The **collabSend** Collaboration defines transactions from the inbound File application to 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 **collabSend**) and click **Next**.

3 For Step 2 or the wizard, from the Web Services Interfaces selection window, double-click **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 selection window, double-click **SeeBeyond** > **eWays** > **EmaileWay** > **EmailClient**. The **EmailClient** OTD is added to the Selected OTDs field.

Figure 29   Collaboration Definition Wizard (Java) - Select Web Service

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

Creating the collabReceive Java Collaboration

The **collabReceive** Collaboration (Java) defines transactions made from the inbound eMail eWay application to the Outbound File eWay and File 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 **collabReceive**) and click **Next**.

3 For Step 2 of the wizard, from the Web Services Interfaces selection window, double-click **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 **SeeBeyond** > **eWays** > **File** > **FileClient**. The Selected OTDs field now lists the **FileClient** OTD.

5 Click **Finish**. The Collaboration Editor with the new JMS2File_Collab Collaboration appears.

5.5.4. 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 collabSend Business Rules

Be careful to open all nodes specified in the directions to connect to the correct item. The **collabSend** Collaboration contains the Business Rule displayed in Figure 30.

Figure 30   collabSend Collaboration Business Rules

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

1 From the Project Explorer tree, double-click **collabSend** to open the Collaboration Editor (Java) to the e-Mailjms_Collab 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 Double-click the **EmailClient_1** and **input nodes** in the Business Rules Designer panes to expand the nodes.

4 To create the **Copy input.Text to EmailClient_1.EmailMessage.MsgText** rule do the following:

A Map **Text** under **input** in the left pane of the Business Rules Designer, to **MsgText** under **EmailClient_1** > **EmailMessage** in the right pane of the Business Rules Designer. To do this, click on **Text** in the left pane of the Business Rules Designer, and drag the cursor to MsgText in the right pane of the Business Rules Designer. A line now connects the two nodes (see Figure 31).

**Figure 31** collabSend - Copy input.Text to EmailClient_1.EmailMessage.MsgText
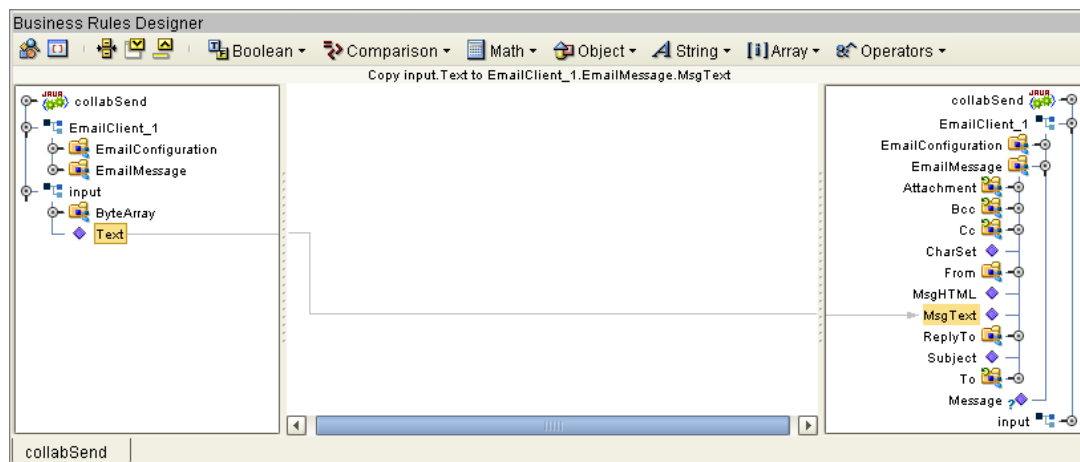
5 To create the **Copy "maileway@thalia.stc.com" to EmailClient_1.EmailMessage.To[0].Address** rule do the following:

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 **EmailMessage** and click **Select method to call,** from the shortcut menu. From the method selection window, select **getTo(int Arg0)**. The **getTo** method box appears.

C From the Business Rules Designer toolbar, click **Math** and select **Literal Number** from the Math menu. The **Number** literal box appears.

D From the Business Rules Designer toolbar, click **Class Browser**. The Class Browser dialog box appears. With **EmailAddress** selected in the All Classes field, double click **setAddress(String arg0)** in the eMailAddress field. The **setAddress** method box appears.

E From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **maileway@"*your send address*"** as the value.

F Map the **0** output node of the Number literal box to the **arg0 (int)** input node of the getTo method box. To do this double-click the 0 value in the Number literal

box, then drag the **0** output node to the **arg0 (int)** input node of the getTo method box.

G   Map the **result (EmailAddress)** output node of the getTo method box to the **EmailAddress** input node of the setAddress method box.

H   Map the **maileway@"*your send address*"** output node of the String literal box to the **arg0 (String)** input node of the setAddress method box (see Figure 32).

Figure 32   Copy "maileway@thalia.stc.com" to EmailClient_1.EmailMessage.To[0].Address
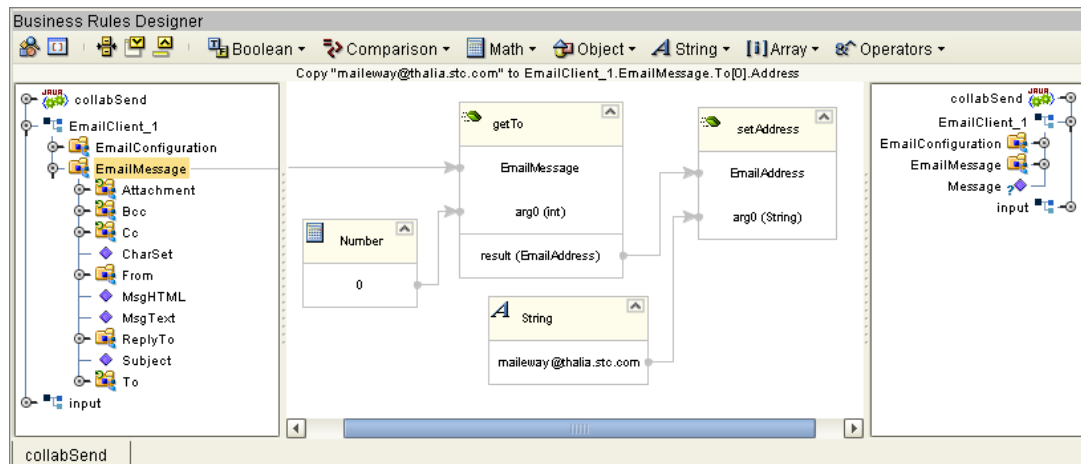
6   To create the **Copy "maileway" to EmailClient_1.EmailMessage.To[0].Name** rule do the following:

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 **EmailMessage** and click **Select method to call,** from the shortcut menu. From the method selection window, select **getTo(int Arg0)**. The **getTo** method box appears.

C   From the Business Rules Designer toolbar, click **Math** and select **Literal Number** from the Math menu. The **Number** literal box appears.

D   From the Business Rules Designer toolbar, click **Class Browser**. The Class Browser dialog box appears. With **EmailAddress** selected in the All Classes field, double click **setName(String arg0)** in the eMailAddress field. The **setName** method box appears.

E   From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **maileway** as the value.

F   Map the **0** output node of the Number literal box to the **arg0 (int)** input node of the getTo method box.

G   Map the **result (EmailAddress)** output node of the getTo method box to the **EmailAddress** input node of the setAddress method box.

H   Map the **maileway** output node of the String literal box to the **arg0 (String)** input node of the setName method box (see **Figure 33 on page 56**).

Figure 33   collabSend - Copy "maileway" to EmailClient_1.EmailMessage.To[0].Name
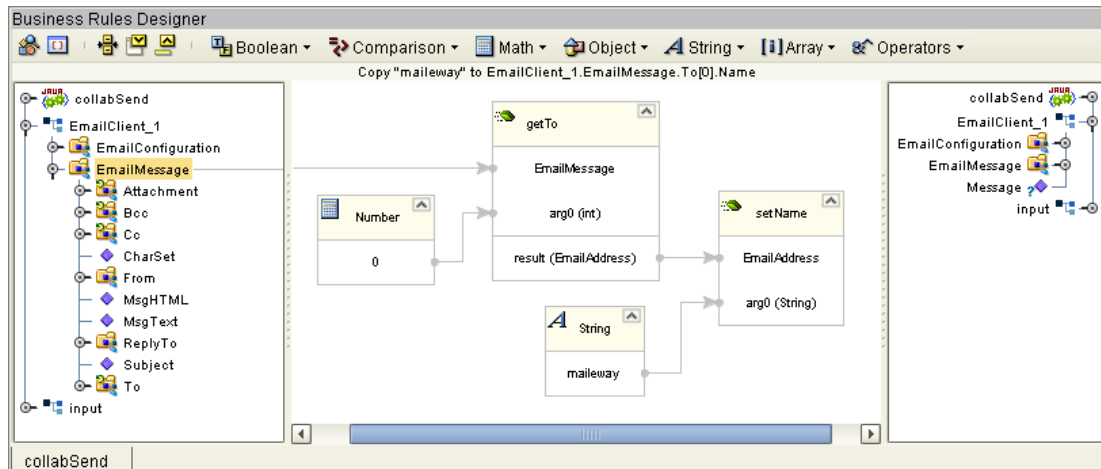
7   To create the **Copy "maileway@thalia.stc.com" to EmailClient_1.EmailMessage.From.Address** rule do the following:

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

B   From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **maileway@"*your from address*"** as the value.

C   Map the **maileway@"*from address*"** output node of the **String** literal box to **Address** under **EmailClient_1** > **EmailMessage** > **From** in the right pane of the Business Rules Designer.

8   To create the **Copy "maileway" to EmailClient_1.EmailMessage.From.Name** rule do the following:

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

B   From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **maileway** as the value.

C   Map the **maileway** output node of the **String** literal box to **Name** under **EmailClient_1** > **EmailMessage** > **From** in the right pane of the Business Rules Designer.

9   To create the **Copy "eMail eWay" to EmailClient_1.EmailMessage.Subject** rule do the following:

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

B   From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **eMail eWay** as the value.

C   Map the **eMail eWay** output node of the Literal method box to **Subject** under **EmailClient_1** > **EmailMessage** in the right pane of the Business Rules Designer.

10   To create the **EmailClient_1.sendMessage** Business Rule do the following:

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. From the method selection window, select **sendMessage()**. The **sendMessage** method box appears (see Figure 34).

**Figure 34** Collaboration Editor (Java) - EmailClient_1.sendMessage
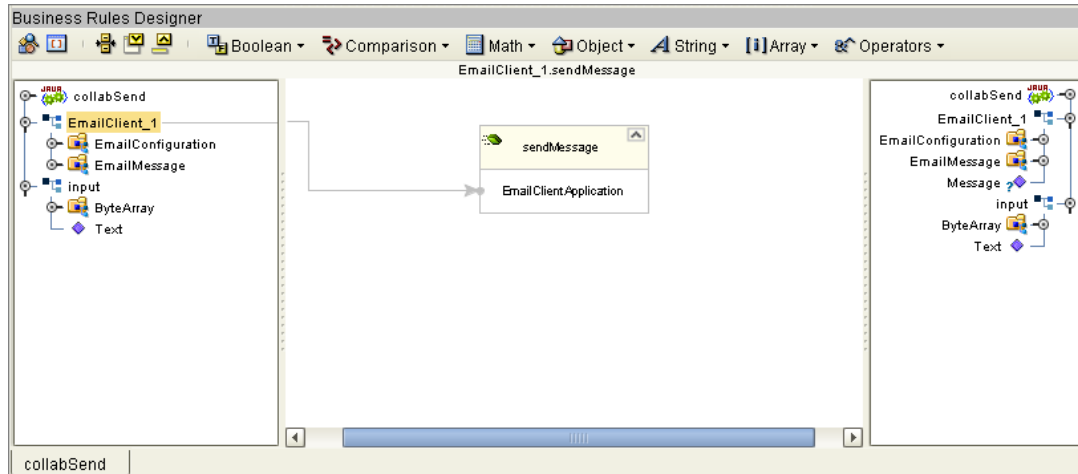
11 From the editor's toolbar, click **Validate** to check the Collaboration for errors.

12 Save your current changes to the repository.

## Creating the collabReceive Collaboration Business Rules

Be careful to open all nodes specified in the directions to connect to the correct item. The e-Mailjms_Collab Collaboration contains the Business Rule displayed in Figure 35

Figure 35 collabReceive Collaboration Business Rules
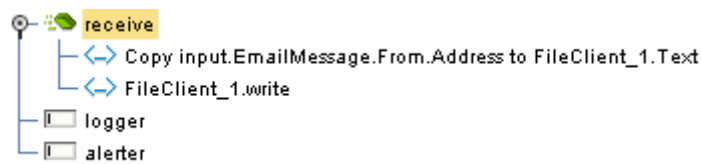
To create the **collabReceive** Collaboration Business Rules do the following:

1 From the Project Explorer tree, double-click **collabReceive** to open the Collaboration Editor (Java) to the **collabReceive** Collaboration.

2 To create the **Copy input.EmailMessage.From.Address to FileClient_1.Text** rule do the following:

A Select the **New rule** in the Business Rules pane.

B Map **Address** under **input** > **EmailMessage** > **From** in the left pane of the Business Rules Designer to **Text** under **FileClient_1** in the right pane of the Business Rules Designer.

3 To create the **FileClient_1.write** Business Rule do the following:

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 the **write** method. The **write** method box appears in the Transformation Designer canvas.

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

5 Save your current changes to the repository.

For more information on how to create Business Rules using the Collaboration Editor see the *eGate Integrator User's Guide*.

5.5.5. Binding the eWay Components

After the Collaborations have been written, the components are associated and Bindings are created in the Connectivity Map.

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

2 Drag and drop the **collabSend** Collaboration from the Project Explorer to the **collabSend1** Service. If the Collaboration was successfully associated, the Service's "gears" icon changes from red to green (see Figure 36).

Figure 36 Connectivity Map - Binding the Components
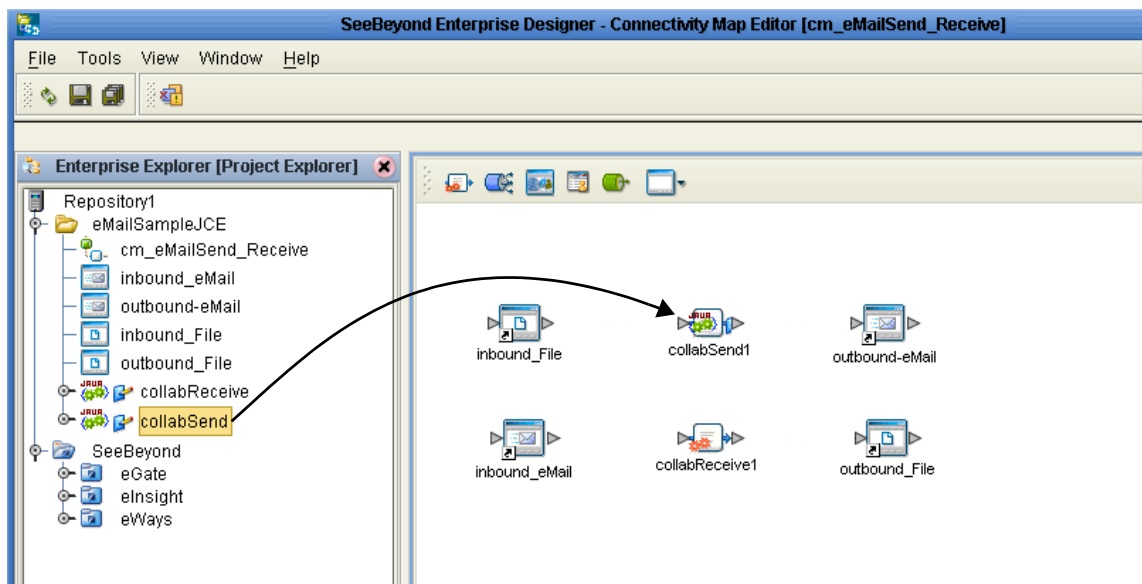
3 Drag and drop **collabReceive** from the Project Explorer to the **collabReceive**1 Service.

4 From the Connectivity Map canvas, double-click **collabSend1**. The **collabSend1** binding dialog box appears with the **collabSend** Rule.

5   From the **collabSend1** binding dialog box, map **FileClient Input** (under Implemented Services) to the inbound **inbound_File** External Application.

6   From the **collabSend1** binding dialog box, map **EmailClient** (under Invoked Services) to the **outbound_eMail1** External Application.

Figure 37   Connectivity Map - Associating (Binding) the Project's Components
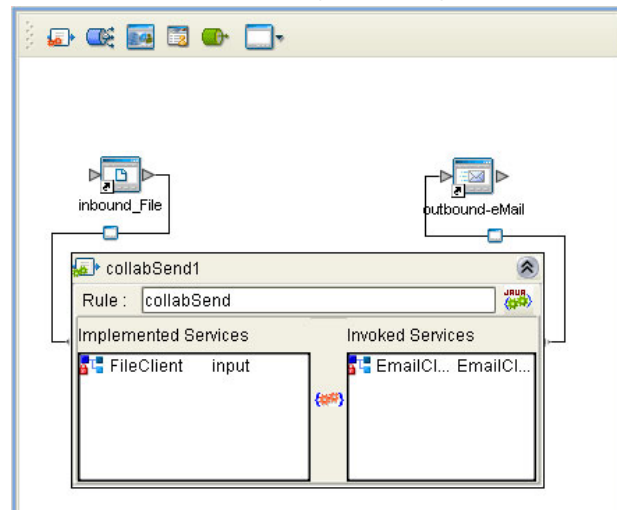
7   Minimize the **collabSend1** binding dialog box by clicking the chevrons in the upper-right corner.

8   From the Connectivity Map, double-click the **collabReceive1** Service. The **collabReceive1** binding dialog box appears with the **collabReceive** Rule.

9   From the **collabReceive1** binding dialog box, drag **EmailClient input** (under Implemented Services) to **inbound_eMail**.

10   From the **collabReceive1** binding dialog box, drag **FileClient** (under Invoked Services) to the outbound **outbound_File** Application.

11   Minimize the **collabReceive1** binding dialog box, and save your current changes to the Repository

5.5.6. 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 **eMailSampleJCE_ENV**.

4   Right-click **eMailSampleJCE_ENV** and select **New e-Mail External System**. Name the External System **EmailExtSysIn** and select Inbound Email eWay as the External System Type. Click **OK**. **EmailExtSysIn** is added to the Environment Editor.

5   Right-click **eMailSampleJCE_ENV** and select **New e-Mail External System**. Name the External System **EmailExtSysOut** and select Outbound Email eWay as the External System Type. Click **OK**. **EmailExtSysOut** is added to the Environment Editor.

6   Right-click **eMailSampleJCE_ENV** and select **New File External System**. Name the External System **FileExtSysIn** and select **Inbound File eWay** as the External System Type. Click **OK**. **FileExtSysIn** is added to the Environment Editor.

7   Right-click **eMailSampleJCE_ENV** and select **New File External System**. Name this External System **FileExtSysOut** and select Outbound File eWay as the External System Type. **FileExtSysOut** is added to the Environment Editor.

8   Right-click **eMailSampleJCE_ENV** and select **New Logical Host**. **LogicalHost1** is added to the Environment Editor.

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

10 Save changes to the repository. The Environment Explorer and Environment Editor now appear as displayed in Figure 38.

Figure 38   Environment Editor - eMailSampleJCE_Env
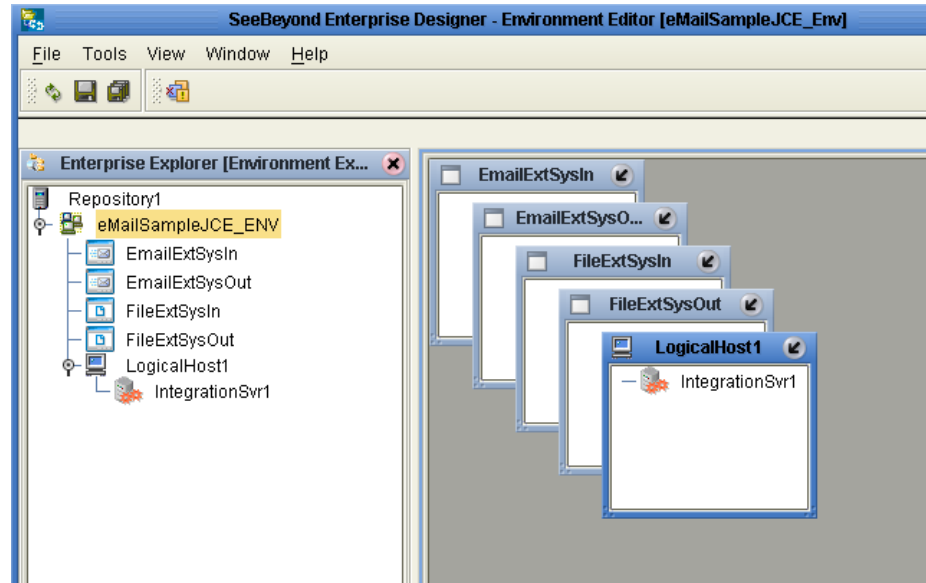
11 Save your current changes to the Repository.

5.5.7. Configuring the eWays

The eMailSampleJCE Project uses four eWays, each represented in the Connectivity Map 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.

The File eWay properties are configured from the Connectivity Map. The e-Mail eWay properties are set from both the Project Explorer's Connectivity Map and the Environment Explorer tree.

Configuring the File eWay Properties

1   Double-click the inbound **File eWay** (see Figure 39), select **Inbound File eWay** in the Templates dialog box and click **OK**.
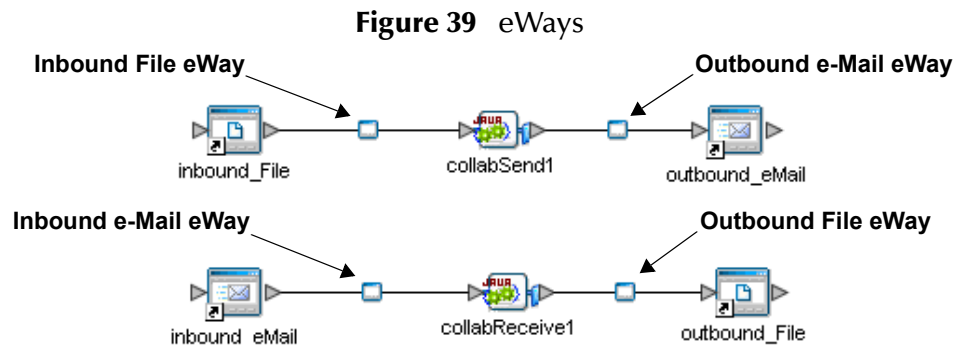
Figure 39   eWays

2   The **Properties Editor** opens to the inbound File eWay properties. Modify the properties for your system, including the settings for the inbound File eWay in Table 7, and click **OK**.

Table 7   Inbound File eWay Settings

Inbound eWay Connection Parameters	
Directory	C:/temp
Input file name	Input*.txt
Polling interval	60000

3   In the same way, modify the outbound File eWay properties for your system, including the settings in Table 8, and click **OK**.

Table 8   Outbound File eWay Settings

Outbound eWay Connection Parameters	
Directory	C:/temp

Configuring the e-Mail eWay Properties

The e-Mail eWay properties must be set in both the Connectivity Map and the Environment Explorer. For more information on the e-Mail eWay properties and the Properties Editor, see **"Configuring the e-Mail eWay Properties" on page 14** or see the *eGate Integrator User's Guide*.

Modifying the e-Mail eWay Connectivity Map Properties

1 From the **Connectivity Map**, double-click the inbound **e-Mail eWay**, select **Inbound Email eWay** in the Templates dialog box and click **OK**. The Properties Editor opens to the Inbound e-Mail eWay properties.

2 Modify the Inbound e-Mail eWay Connectivity Map properties for your system, including the settings in **Table 9 on page 62**, and click **OK**.

Table 9   Inbound e-Mail eWay Connectivity Map Properties

Inbound e-Mail eWay (Connectivity Map) Properties	
Polling Setting Set as directed, otherwise use the default settings	
Polling Interval	60000

3 The **Outbound e-Mail eWay** Connectivity Map properties contain only SSL related parameters. Use the default for these settings.

Modifying the e-Mail eWay Environment Explorer Properties

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

2 Modify the inbound e-Mail eWay environment properties for your system, including the settings in Table 10, and click **OK**.

Table 10   Inbound e-Mail Environment Explorer eWay Properties

Inbound e-Mail eWay Environment Properties	
Connection Settings Set as directed, otherwise use the default settings	
Host Receive	*Host name of the receiving POP3 server*
Password Receive	*User password for receiving e-mail*
Port Receive	*TCP/IP Port Number for receiving e-mail*
User Receive	*User name for receiving e-mail*

1 From the **Environment Explorer** tree, right-click the outbound e-Mail eWay external system (**EmailExtSysOut** in this sample), and select **Properties** from the shortcut menu. The Properties Editor appears.

2 Modify the outbound e-Mail eWay environment properties for your system, including the settings in Table 11, and click **OK**.

Table 11 Outbound e-Mail Environment Explorer eWay Properties

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

5.5.8 Creating and Activating the Deployment Profile

Deployment Profiles are specific instances of a Project in a particular Environment. A Deployment Profile contains information about 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.

To create the **eMailSampleJCE_DP** Deployment Profile do the following:

1 From the Enterprise Explorer's Project Explorer, right-click the Project (**eMailSampleJCE**) and select **New** > **Deployment Profile** from the shortcut menu.

2 Enter a name for the Deployment Profile (for this sample **eMailSampleJCE_DP**). Make sure that the selected Environment is **eMailSampleJCE_ENV**. Click **OK**. The Deployment Profile Editor appears.

3 Click the **Auto Map** icon as displayed in **Figure 40 on page 64**. The Project's components are automatically mapped to their system windows as seen in **Figure 41 on page 64**. If any of the Project components are not mapped automatically after Auto Map is used, those component can be mapped manually by following the appropriate steps below. Once all components are mapped, proceed to step 9.

Figure 40 Deployment Profile - Auto Map
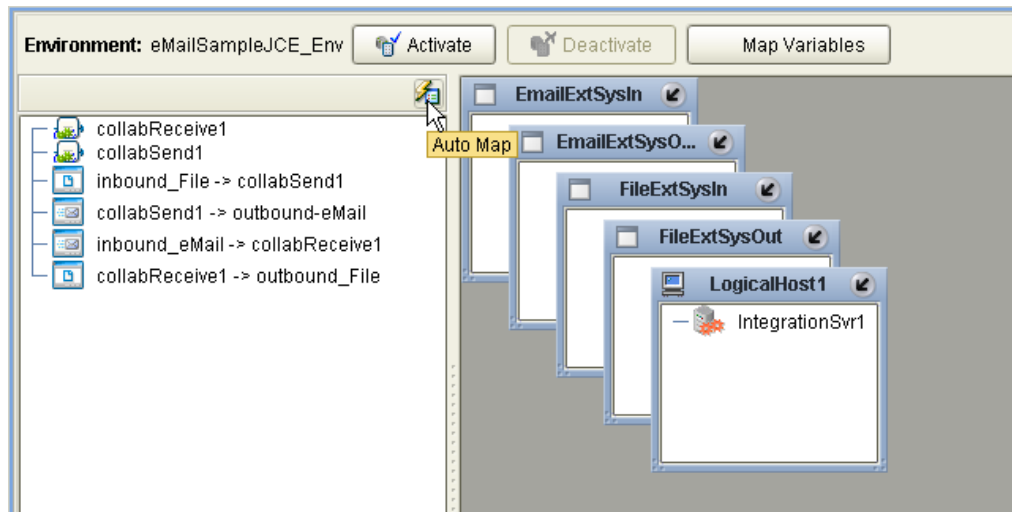
4   From the left pane of the Deployment Editor, drag the **inbound_File -> collabSend1** (External Application) object to the **FileExtSysIn** window.

5   Drag the **collabReceive1 -> outbound_File** (External Application) object to the **FileExtSysOut** window.

6   Drag the **collabSend1 -> outbound_eMail** (External Application) object to the **EmailExtSysOut** window.

7   Drag the **inbound_eMail -> collabSend1** (External Application) object to the **EmailExtSysIn** window.

8   Drag the **collabSend1** and **collabReceive1** (Services) to **IntegrationSvr1** in the **LogicalHost1** window (see Figure 41).

Figure 41 Deployment Profile
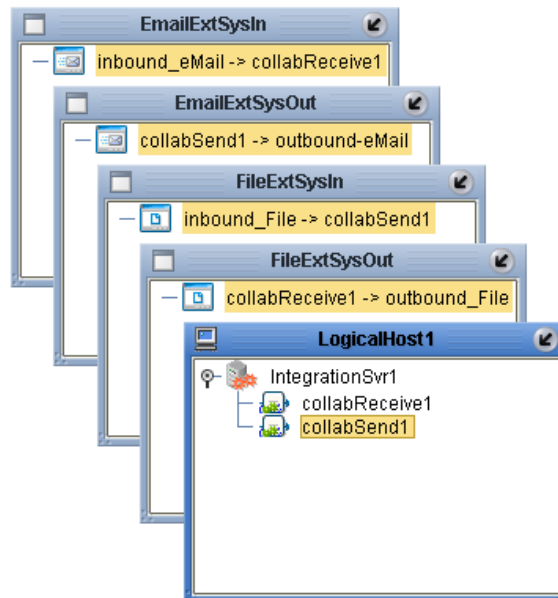
9   Click **Activate**. When activation succeeds, save the changes to the Repository.

5.5.9. **Running the Project**

The following directions assume that the Enterprise Designer was downloaded to **C:\ican50**. If this is not the case, replace that location in the following directions with the appropriate location.

1 From the Enterprise Manager Downloads tab, download **Logical Host - for win32**.

2 Extract the file to the **ican50\LogicalHost2** directory. You must specify the **LogicalHost2** directory for it to be created.

3 Navigate to **C:\ican50\LogicalHost2\logicalhost\bootstrap\config** directory and open the **logical-host.properties** file using Notepad™.

4 Enter the following information in the appropriate fields:

   ♦ Logical Host root directory: **ican50\LogicalHost2\logicalhost**

   ♦ Repository URL: **http://localhost:***port number/repository name*

   ♦ Repository user name and password: *Your user name and password*

   ♦ Logical Host Environment name: **eMailSampleJCE_Env**

   ♦ Logical Host name: **LogicalHost2**

   Save your changes to **logical-host.properties** and close the file.

5 Run the **bootstrap.bat** file in the **ican50\LogicalHost2\logicalhost\bootstrap\bin** directory.

6 Copy the sample input data file to the input directory.

5.6 **Alerting and Logging**

eGate provides an alerting and logging feature. This allows monitoring of messages, and captures any adverse messages in order of severity based on configured severity level and higher. To enable Logging, please see the *eGate Integrator User's Guide*.

Note: *The alerts/status notifications for the e-Mail eWay are currently limited to Started, Running, Stopping, and Stopped.*

5.7 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).

In order for e-mail to be in encoded to support Japanese character text, the value for **CharSet** must be set to **ISO-2022-JP**. To enable Japanese character support for sent or received messages, do the following:

1 Upload the **charsets.jar** file as follows:

   A From the Enterprise Designer's Environment Explorer, right-click your Environments LogicalHost and select **Upload File** from the shortcut menu. The **Upload Third Party Files** dialog box appears.

   B From the **Upload Third Party Files** dialog box, click **Add**. Select **charsets.jar**, located in the ***<ican50>\<logicalhost>\jre\lib*** directory, where <ican50> is your ICAN installation, and <logicalhost> is your LogicalHost directory. The charsets.jar file now appears in the Upload Third Party Files dialog box (see Figure 42). Click **OK** to close the dialog box.

   C Right click the Logical Host and select **Apply** from the menu.

Note: *Your installation may have automatically added the JAR file to your* ***logicalhost/jre/lib*** *folder. Check this location on your system. If charset.jar is already located there, you can disregard the directions in step 1 .*

If the Logical Host is running, it must be recycled to pickup the new JAR file.

Figure 42 Enterprise Explorer - Upload charsets.jar
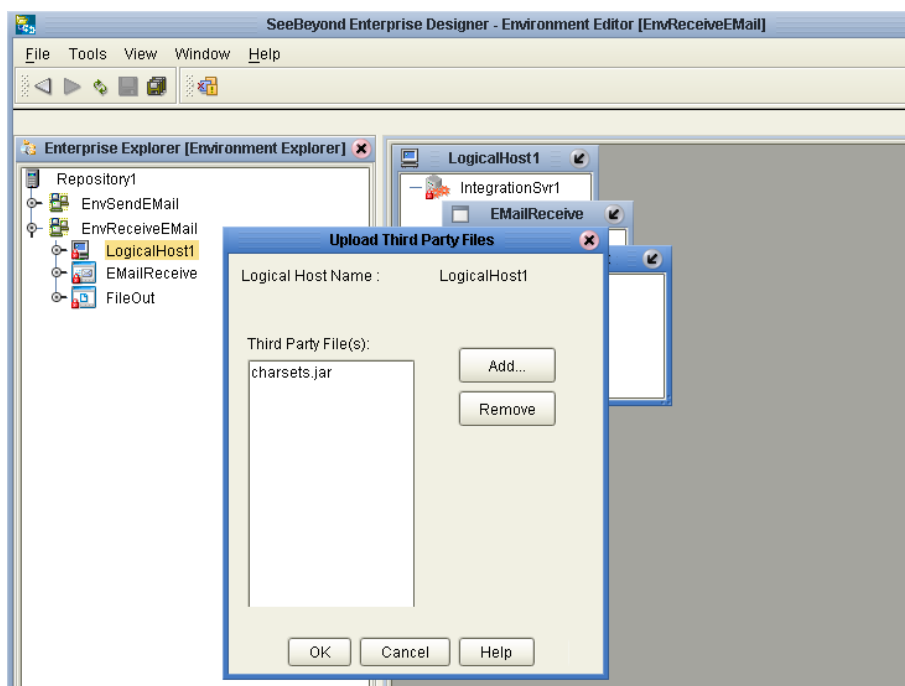
2 Set the ISO-2022-JP character text to CharSet as follows:

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

B From the Business Rules Designer toolbar, click **String** and select **Literal String** from the String menu. The **String** literal box appears. Enter **ISO-2022-JP** as the value.

C Map the **ISO-2022_JP** output node of the literal **String** method box to **CharSet** under **EmailClient_1** > **EmailMessage** (for send) in the right pane of the Business Rules Designer (see **Figure 43 on page 67**). In the "receive" Collaboration, **EmailMessage** is under the **Input** node.

Figure 43 Business Rule Designer - Japanese Character Encoding
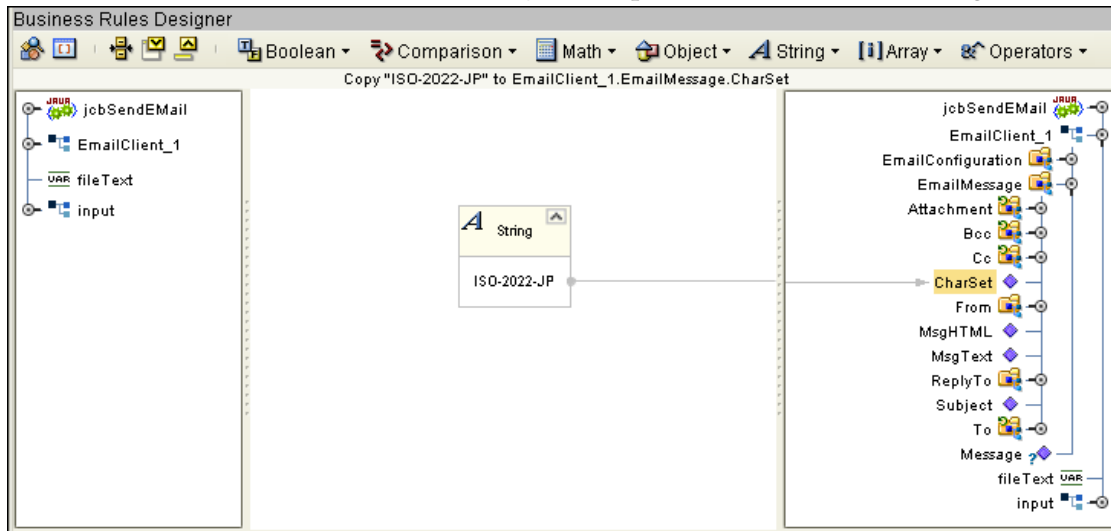
This adds the following Java code to the "Send" Collaboration:

```
EmailClient_1.getEmailMessage().setCharSet( "ISO-2022-JP" );
```

or to the "Receive" Collaboration:

```
input.getEmailMessage().setCharSet( "ISO-2022-JP" );
```

Japanese Character Support in the EmailMessage OTD

The EmailMessage OTD is the user interface for the e-Mail header. The following fields of the EmailMessage OTD allow Japanese Character support as follows:

▪ When the OTD CharSet is set to **ISO-2022-JP**, the Strings set in the **EmailMessage** OTD by the following methods:

◆ **EmailAddress.SetName(String)** for To/cc/Bcc of the EmailMessage

◆ **EmailMessage.setSubject(String)**

◆ **EmailAttachment.setName(String)**

are sent in the e-mail header in **ISO-2022-JP** encoding.

- The message content stored using **EmailMessage.setMsgText(String)** or **EmailMessage.setHTML(String)** is stored in **ISO-2022-JP** encoding in the e-mail message sent by the eWay.

- The attachment data set by **EmailAttachmment.setContents(byte[])** is not modified by the eWay. Therefore, the user data must be encoded in **ISO-2022-JP** in the byte array before the attachment content is set.

Note: *There is a problem in 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).*

Using SSL

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

What's in This Chapter

- **TrustStores** on page 70
- **SSL Handshaking** on page 71

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 44).

Figure 44 General SSL Operation
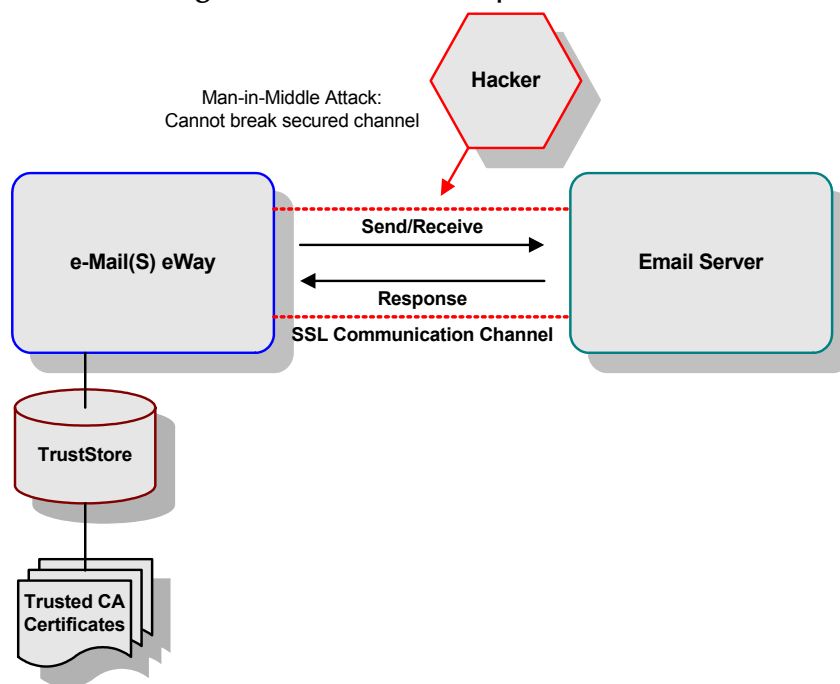
This SSL feature is supported through the use of JSSE version 1.0.3.

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.

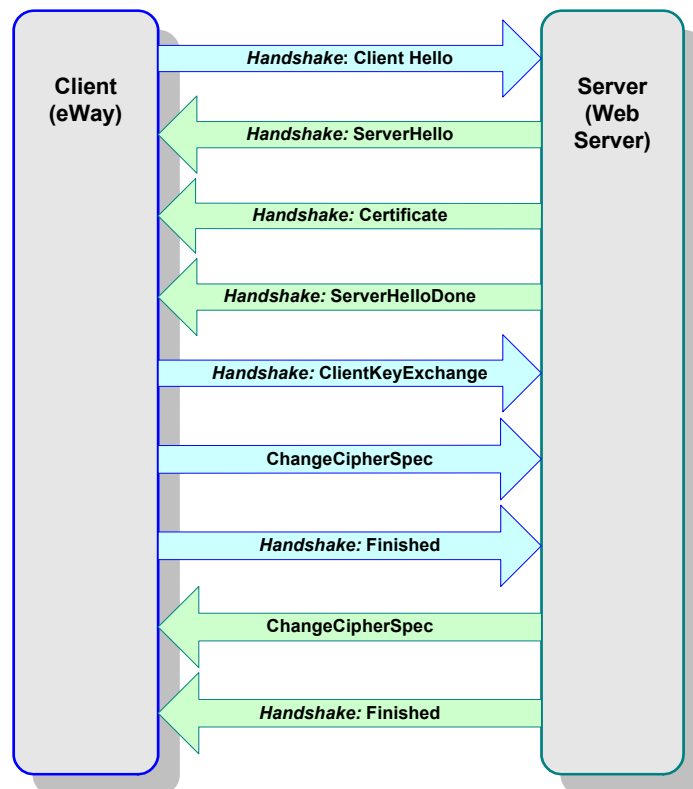If you are importing certificates into an existing TrustStore, use the following:

```
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 45.

Figure 45 Server-side Authentication

Index