

***SeeBeyond ICAN Suite***

# eGate Integrator Tutorial

*Release 5.0.4*



The information contained in this document is subject to change and is updated periodically to reflect changes to the applicable software. Although every effort has been made to ensure the accuracy of this document, SeeBeyond Technology Corporation (SeeBeyond) assumes no responsibility for any errors that may appear herein. The software described in this document is furnished under a License Agreement and may be used or copied only in accordance with the terms of such License Agreement. Printing, copying, or reproducing this document in any fashion is prohibited except in accordance with the License Agreement. The contents of this document are designated as being confidential and proprietary; are considered to be trade secrets of SeeBeyond; and may be used only in accordance with the License Agreement, as protected and enforceable by law. SeeBeyond assumes no responsibility for the use or reliability of its software on platforms that are not supported by SeeBeyond.

SeeBeyond, e\*Gate, e\*Way, and e\*Xchange are the registered trademarks of SeeBeyond Technology Corporation in the United States and/or select foreign countries. The SeeBeyond logo, SeeBeyond Integrated Composite Application Network Suite, eGate, eWay, eInsight, eVision, eXchange, eView, eIndex, eTL, ePortal, eBAM, and e\*Insight are trademarks of SeeBeyond Technology Corporation. The absence of a trademark from this list does not constitute a waiver of SeeBeyond Technology Corporation's intellectual property rights concerning that trademark. This document may contain references to other company, brand, and product names. These company, brand, and product names are used herein for identification purposes only and may be the trademarks of their respective owners.

© 2004 by SeeBeyond Technology Corporation. All Rights Reserved. This work is protected as an unpublished work under the copyright laws.

**This work is confidential and proprietary information of SeeBeyond and must be maintained in strict confidence.**

Version 20040630145255.

# Contents

---

## Chapter 1

<b>Introduction</b>	<b>6</b>
Organization of Information	6
Supporting Documents	6
Writing Conventions	7
Additional Conventions	7
ICAN Suite Installation Requirements	8
SeeBeyond Web Site	8

---

## Chapter 2

<b>Features of the Enterprise Designer</b>	<b>9</b>
Enterprise Designer Components	9
Menu Bar	10
Enterprise Explorer	11
Project Editor	11

---

## Chapter 3

<b>Building a Project - Scenario1</b>	<b>12</b>
This chapter includes	12
<b>Business Challenge</b>	<b>12</b>
<b>Project Overview</b>	<b>13</b>
Project Description	13
Process Overview	13
<b>ICAN Files Required for the Tutorials</b>	<b>16</b>
Repository, Logicalhost and Environment	16
File Names	16
<b>Sample Data</b>	<b>16</b>
Obtain Sample Data Files	16
Create Sample Input XML File	17
Create Input (DTD) File	18

Create Output (DTD) File	18
<b>Create New Project</b>	<b>19</b>
<b>Create Connectivity Map</b>	<b>20</b>
Add Components to the Connectivity Map	22
<b>Create New Object Type Definition</b>	<b>24</b>
<b>Configure the Services</b>	<b>28</b>
Create Service1	28
<b>Apply the Business Rules</b>	<b>33</b>
<b>Apply the Collaborations</b>	<b>40</b>
Link Components in the Connectivity Map	40
<b>Configure the eWays</b>	<b>41</b>
Multiple records per file (True or False)	44
<b>Create Environment</b>	<b>44</b>
Create Environment and Logical Host	44
<b>Create and Activate Deployment Profile</b>	<b>47</b>
Deploy Project1	48
<b>Run the Bootstrap and Management Agent</b>	<b>49</b>
<b>Run the Bootstrap</b>	<b>50</b>
<b>Run Project1</b>	<b>51</b>
<b>Verify Output Data</b>	<b>53</b>
Edit Input File in Real Time	54

---

## Chapter 4

<b>Building a Project - Scenario2</b>	<b>55</b>
Required Resources	55
This chapter includes	55
<b>Business Challenge</b>	<b>56</b>
<b>Project Overview</b>	<b>57</b>
Naming conventions used in this scenario	57
Project Description	57
<b>Sample Data</b>	<b>60</b>
Create Sample Input XML File	60
Create Input (DTD) File	60
Create Output (DTD) File	61
<b>Create New Project</b>	<b>61</b>
<b>Create Connectivity Map</b>	<b>63</b>
<b>Add Components to the Connectivity Map</b>	<b>65</b>
Add Components to the Connectivity Map	65
<b>Create New Object Type Definition</b>	<b>67</b>

Configure Services for Input	72
Configure svcTimecard_to_topic	72
Apply Business Rules for jcdTimecardToTopic_collab	76
Configure Services for Output	77
Apply Business Rules for jcdTopicToPayrollOut_collab	81
Apply Collaborations and Link Components	90
Link Components in the Connectivity Map	90
Configure eWays and JMS Connections	92
JMS Client Configuration	95
Create Environment	96
Create and Activate Deployment Profile	98
Run Bootstrap and Management Agent	100
Run Project2	100
Verify Output Data	101

---

## Chapter 5

<b>Enterprise Manager Overview</b>	<b>103</b>
Browser-based monitor console	103
When to Use the Monitor	103
Using Project View	103
Using Environment View	104
Locating the Problem Node	104
Starting and Using Enterprise Manager	104
Monitor a Collaboration in your Connectivity Map	104
Stop Service	105
Monitor Alerts	106
Check Log Messages	107
<b>Glossary</b>	<b>109</b>
e*Gate 4.x Terms in eGate 5.0	112
<b>Index</b>	<b>114</b>

# Introduction

Welcome to the *eGate Integrator Tutorial*. The Tutorial must be run on a Windows system.

This document includes an overview of the Enterprise Designer graphical user interface (GUI) and tutorials to help you learn how to create and run a Project in eGate Integrator.

---

## 1.1 Organization of Information

This document includes the following chapters:

- **Chapter 1 “Introduction” on page 6** overviews the contents of the *eGate Integrator Tutorial*, describes the writing conventions used in this document, and provides a complete list of related eGate Integrator documentation.
- **Chapter 2 “Features of the Enterprise Designer” on page 9** introduces the Enterprise Designer Components, Menu Bar, Enterprise Explorer, and Project Editor.
- **Chapter 3 “Building a Project - Scenario1” on page 12** provides a simple “end-to-end” scenario where you perform a data concatenation.
- **Chapter 4 “Building a Project - Scenario2” on page 55** provides an “end-to-end” scenario where you perform data transformation, involving a multiplication operation.
- **Chapter 5 “Enterprise Manager Overview” on page 103** provides an overview of the Enterprise Manager which is used to view and monitor projects dynamically.
- **“Glossary” on page 109.**
- **“Index” on page 114.**

---

## 1.2 Supporting Documents

The following SeeBeyond documents provide additional information about eGate Integrator:

- *SeeBeyond ICAN Suite Installation Guide*
- *eGate Integrator System Administration Guide*

- *eGate Integrator Release Notes*
- *eGate Integrator User's Guide*
- *Message Server Reference Guide*
- *SeeBeyond ICAN Suite Deployment Guide*
- *SeeBeyond ICAN Suite Primer*

See the *SeeBeyond ICAN Suite Primer* for a complete list of eGate Integrator documentation. You can also refer to the appropriate Windows or UNIX documents, if necessary.

## 1.3 Writing Conventions

The following writing conventions are observed throughout this document.

**Table 1** Writing Conventions

Text	Convention	Example
Button, file, icon, parameter, variable, method, menu, and object names.	<b>Bold</b> text	<ul style="list-style-type: none"> <li>▪ Click <b>OK</b> to save and close.</li> <li>▪ From the <b>File</b> menu, select <b>Exit</b>.</li> <li>▪ Select the <b>logicalhost.exe</b> file.</li> <li>▪ Enter the <b>timeout</b> value.</li> <li>▪ Use the <b>getClassname()</b> method.</li> <li>▪ Configure the <b>Inbound</b> File eWay.</li> </ul>
Command line arguments and code samples	Fixed font. Variables are shown in <b><i>bold italic</i></b> .	<code>bootstrap -p <b><i>password</i></b></code>
Hypertext links	<b>Blue</b> text	<a href="http://www.seebeyond.com">http://www.seebeyond.com</a>

### Additional Conventions

#### Windows Systems

For the purposes of this guide, references to “Windows” will apply to Microsoft Windows Server 2003, Windows XP, and Windows 2000.

#### Path Name Separator

This guide uses the backslash (“\”) as the separator within path names. If you are working on a UNIX system, please make the appropriate substitutions.

---

## 1.4 ICAN Suite Installation Requirements

To simplify these examples, this tutorial assumes you have all of the following eGate components installed on a single Windows system:

- eGate Repository
- Logical Host
- Enterprise Designer
- eGate product
- File eWay product

Refer to the *SeeBeyond ICAN Suite Installation Guide* for system requirements and installation instructions. See also: [Chapter 3, ICAN Files Required for the Tutorials](#) on page 16.

---

## 1.5 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:

<http://www.seebeyond.com>



# Features of the Enterprise Designer

The Enterprise Designer is the graphical user interface (GUI) used to design and implement ICAN 5.0 projects. This chapter overviews the features and interface of the Enterprise Designer window.

This chapter includes

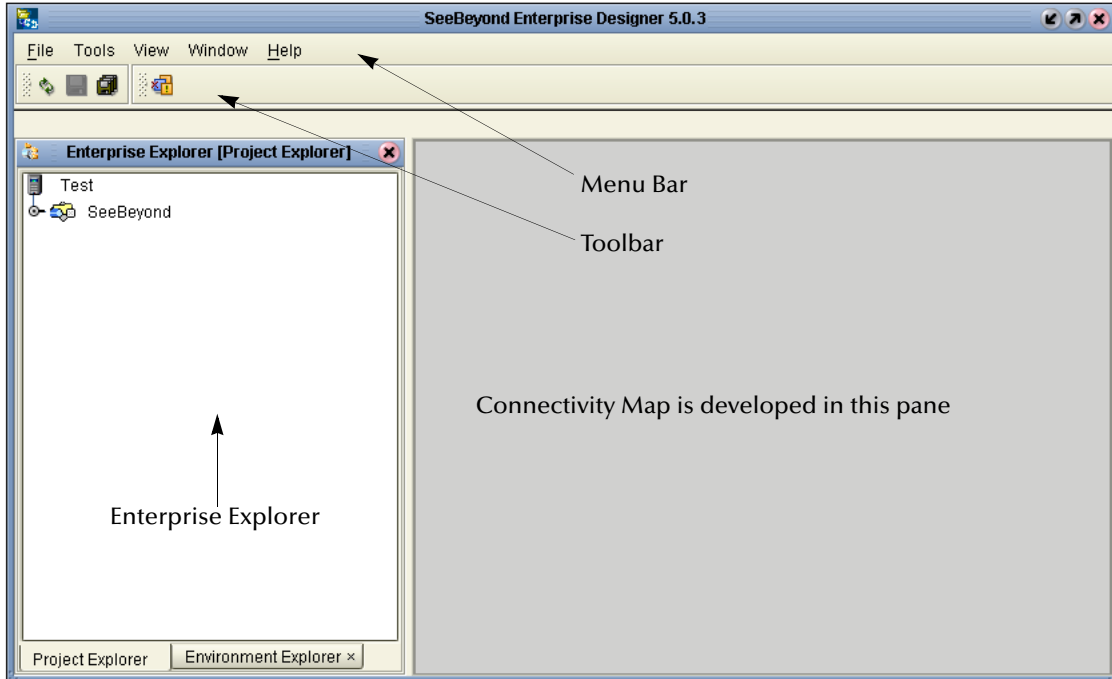
- [“Enterprise Designer Components” on page 9](#)
- [“Menu Bar” on page 10](#)
- [“Enterprise Explorer” on page 11](#)
- [“Project Editor” on page 11](#)

---

## 2.1 Enterprise Designer Components

The Enterprise Designer is used to create and configure the components of an eGate Project. Each component of this interface is identified in Figure 1.

**Figure 1** SeeBeyond Enterprise Designer Window



*Note:* This chapter provides a high-level overview of the Enterprise Designer GUI. Refer to the *eGate Integrator User's Guide* for a more detailed description of the menu bar, toolbar, Enterprise Explorer, and Enterprise Designer.

---

## 2.2 Menu Bar

The menu bar shown below in Figure 2 provides access to a series of menu commands used to build and manage a Project.

**Figure 2** Enterprise Designer Menu Bar

**File Tools View Window Help**

The function of each menu is briefly described below:

- **File**—Lists options for save, save all, and exit.
- **Tools**—Lists options for Impact Analyzer, Update Center and Setup Options.
- **View**—Lists options to view the Environment Explorer and Project Explorer.
- **Window**—Lists options for the various window displays, cascade, tile, etc.
- **Help**—online help for all installed modules

---

## 2.3 Enterprise Explorer

The Enterprise Explorer organizes all of the components of a Project into a series of folders and contains the following two tabs:

- **Project Explorer**—Logical configurations designed to help solve a business problem. This branch includes all the components of an Enterprise Designer Project, including Connectivity Maps, Services, Object Type Definitions (OTD), and Deployment Profiles.
- **Environment Explorer**—Collections of logical hosts and external systems capable of hosting eGate components and information about external systems, which may be involved with an eGate configuration.

---

## 2.4 Project Editor

The Project Editor contains the “nuts-and-bolts” of a Project. This part of the Enterprise Designer is empty when you start a new Project. However, as you work through the tutorial, the Project Editor quickly fills with components and graphical structures representing the various stages of the Project. The types of windows in the Project Editor area include:

- **Connectivity Map**—Contains business logic components, such as Services, Topics, queues, and eWays, that you include in the structure of a Project.
- **Collaboration Editor (Java)**—Contains the business rules defined in Java.
- **Collaboration Editor (XSLT)**—Contains the business rules defined in XSLT.
- **OTD Editor**—Edits and tests the OTDs (Object Type Definitions)
- **Deployment Profile Editor**—Edits the deployment profile
- **Environment**—Collection of physical resources (logical hosts and external systems)

# Building a Project - Scenario1

This tutorial provides step-by-step procedures for creating and testing an eGate Project. This Project addresses a very simple business challenge where an input TimeCard file is read and a workfile is output.

### This chapter includes

- “Business Challenge” on page 12
- “Project Overview” on page 13
- “ICAN Files Required for the Tutorials” on page 16
- “Sample Data” on page 16
- “Create New Project” on page 19
- “Create Connectivity Map” on page 20
- “Add Components to the Connectivity Map” on page 22
- “Create New Object Type Definition” on page 24
- “Configure the Services” on page 28
- “Link Components in the Connectivity Map” on page 40
- “Configure the eWays” on page 41
- “Create Environment” on page 44
- “Run the Bootstrap and Management Agent” on page 49
- “Verify Output Data” on page 53

---

## 3.1 Business Challenge

The Project described in this chapter provides a solution to the following business challenge:

In this very simple scenario **FirstName** and **LastName** from the input file are concatenated into the field **FullName** in the output file. **JobTitle**, **HoursWorked** and **Rate** are disregarded in Scenario1.

- The input data in this system is in XML format with six fields: **EmployeeNumber**, **LastName**, **FirstName**, **JobTitle**, **HoursWorked**, and **Rate**.

- The output workfile in XML format contains the following fields: **FullName**, and **EmpNo**.

---

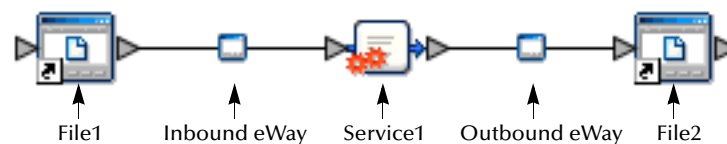
## 3.2 Project Overview

Create a basic eGate Project to meet the business challenge described in **“Business Challenge” on page 12**.

### 3.2.1 Project Description

The finished eGate Project contains components used to move the data throughout the Project and publish the data in the appropriate format for the output workfile.

**Figure 3** Project Connectivity Map



These components perform the following functions:

- **File1:** The Input file *External Application* contains the Input XML data file.
- **Inbound eWay:** The inbound eWay polls a specified location on the file system for the input XML data file.
- **Service1:** This Service performs a concatenation of the name fields.
- **Outbound eWay:** This eWay publishes the finished output data to the file system in the format appropriate for the output work file.
- **File2:** This is the output work file.

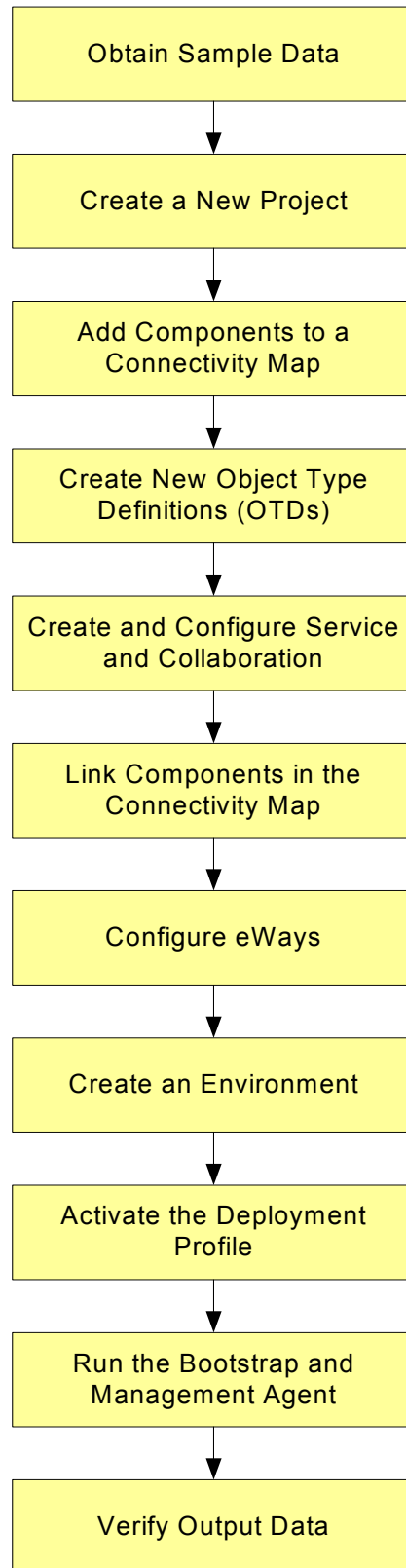
### 3.2.2 Process Overview

Complete the following general steps:

- 1 **“Sample Data” on page 16**
- 2 **“Create New Project” on page 19**
- 3 **“Create Connectivity Map” on page 20**
- 4 **“Add Components to the Connectivity Map” on page 22**
- 5 **“Create New Object Type Definition” on page 24**
- 6 **“Configure the Services” on page 28**

- 7 **"Apply the Business Rules" on page 33**
- 8 **"Apply the Collaborations" on page 40**
- 9 **"Configure the eWays" on page 41**
- 10 **"Create Environment" on page 44**
- 11 **"Run the Bootstrap and Management Agent" on page 49**
- 12 **"Verify Output Data" on page 53**

**Figure 4** Project Diagram



---

## 3.3 ICAN Files Required for the Tutorials

Please refer to the *SeeBeyond ICAN Suite Installation Guide* for instructions about uploading files to the Repository. The following files are required to run the tutorials:

- License.sar
- eGate.sar
- ProductManifest.xml
- FileeWay.sar
- eGateDocs.sar (PDF and sample files)

The Tutorial assumes the software is installed in C:\ican50. Your installation may be different.

Although not required for the Enterprise Manager you will have more features for GUI displays if you use the SVG Viewer. Use the **EnterpriseManager-Plugin-Win98ToXP.sar** (required for the Adobe SVG Viewer plug-in for Windows). Enterprise Manager is not required for the tutorials.

### 3.3.1 Repository, Logicalhost and Environment

You may use the same Repository, Logicalhost and Environment for both tutorials, Scenario1 and Scenario2. You should use different names for the deployments. Scenario1 should be deployed as Deployment1 and Scenario2 should be deployed as Deployment2.

### 3.3.2 File Names

The file names used in Scenario1 (Project1) should be different than the file names used in Scenario2 (Project2). See the following section, Sample Data.

---

## 3.4 Sample Data

This Project uses the sample files found in the Project1 folder of the **eGate Tutorial Sample** that you download by clicking the **Documentation** tab in the Enterprise Manager. The files used in Scenario1 are: **s1input\_multiple.dtd**, **s1output\_multiple.dtd**, and **TimeCard-Multiple.xml**. (The “s1” in the file names stands for Scenario1.)

### 3.4.1 Obtain Sample Data Files

You can download the sample data files from the Enterprise Manager. First open the Enterprise Manager and then perform the following steps.



## Download Sample Files

Make sure you are in the Enterprise Manager.

- 1 Click the **Documentation** tab.
- 2 Under **Products**, click **eGate Integrator**.
- 3 Click **Download eGate Tutorial Sample**.

Save the zip file to a location on your computer.

*Note:* You can also create the files you need. See sections 3.4.2, 3.4.3, and 3.4.4 below.

## 3.4.2 Create Sample Input XML File

Use a text editor such as Microsoft Notepad to create the input XML file. You can also copy and paste from this document (PDF) using the text select tool from within Adobe Acrobat.

### Create XML file

Your XML file should look similar to the structure and syntax below. In this scenario of the tutorial there are four records.

```
<?xml version="1.0" encoding="UTF-8" ?>
  <Employees>
    <Employee>
      <EmployeeNumber>100</EmployeeNumber>
      <LastName>Smith</LastName>
      <FirstName>Jane</FirstName>
      <JobTitle>Manager</JobTitle>
      <HoursWorked>40</HoursWorked>
      <Rate>55</Rate>
    </Employee>
  <Employee>
    <EmployeeNumber>123</EmployeeNumber>
    <LastName>Chang</LastName>
    <FirstName>Ling</FirstName>
    <JobTitle>Manager</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>60</Rate>
  </Employee>
  <Employee>
    <EmployeeNumber>118</EmployeeNumber>
    <LastName>Jamison</LastName>
    <FirstName>Dick</FirstName>
    <JobTitle>Clerk</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>22</Rate>
  </Employee>
  <Employee>
    <EmployeeNumber>144</EmployeeNumber>
    <LastName>Nakayama</LastName>
    <FirstName>Takeshi</FirstName>
    <JobTitle>Supervisor</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>35</Rate>
  </Employee>
</Employees>
```

- 1 In Windows Explorer, create a directory for the data files. For this scenario, use **c:\eGateData\Project1**.
- 2 In Notepad, create an XML file containing the data shown above.

- 3 Save the file as **TimeCard\_Multiple.xml** in the **c:\eGateData\Project1** directory.

### 3.4.3 Create Input (DTD) File

The **s1input\_multiple.dtd** file is used as a representation of the data in the TimeCard system.

The **s1input\_multiple.dtd** file uses the following elements:

- **EmployeeNumber**
- **LastName**
- **FirstName**
- **JobTitle**
- **HoursWorked**
- **Rate**

#### Create Input (DTD) File

Your input (DTD) should look similar to the structure and syntax below.

- 1 In Notepad, create an XML file containing the following information:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Employees(Employee+)>
<!ELEMENT Employee(EmployeeNumber, LastName, FirstName, JobTitle,
HoursWorked, Rate)>
<!ELEMENT EmployeeNumber(#PCDATA)>
<!ELEMENT LastName(#PCDATA)>
<!ELEMENT FirstName(#PCDATA)>
<!ELEMENT JobTitle(#PCDATA)>
<!ELEMENT HoursWorked(#PCDATA)>
<!ELEMENT Rate(#PCDATA)>
```

- 2 Save the file as **s1input\_multiple.dtd** in the **c:\eGateData\Project1** directory.

### 3.4.4 Create Output (DTD) File

The **s1output\_multiple.dtd** file is used as a representation of the data in the output workfile.

The **s1output\_multiple.dtd** file uses the following elements:

- **FullName**
- **EmpNo**

#### Create Output (DTD) file

Your output (DTD) should look similar to the structure and syntax below.

- 1 In Notepad, create an XML file containing the following information:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Employees(Employee+)>
<!ELEMENT Employee(FullName, EmpNo)>
<!ELEMENT FullName(#PCDATA)>
<!ELEMENT EmpNo(#PCDATA)>
```

- 2 Save the file as `s1output_multiple.dtd` in the `c:\eGateData\Project1` directory.

---

## 3.5 Create New Project

There are many ways to create a Project and link the modules. This tutorial presents one scenario. Begin by creating and naming a Project in the Enterprise Designer.

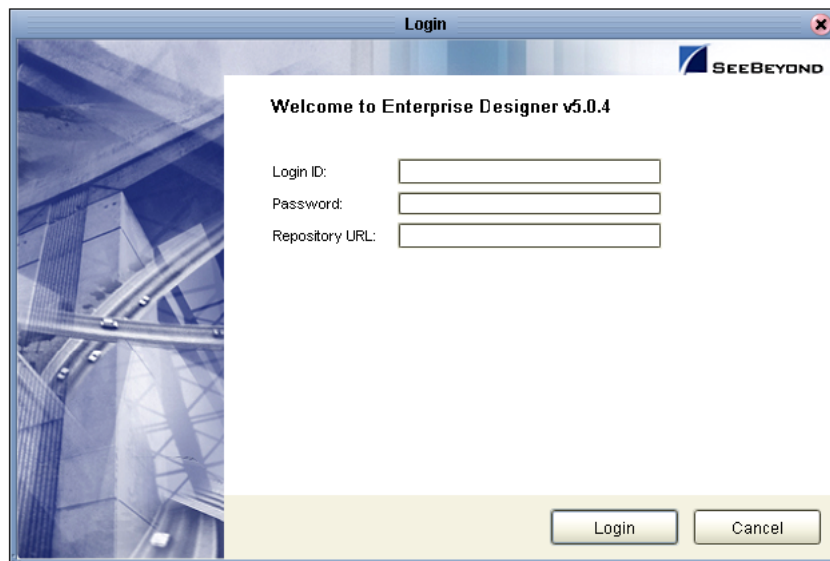
### Start Enterprise Designer

Start the Enterprise Designer according to your organization's instructions.

- 1 Start the Enterprise Designer by executing `runed.bat` in your `ican50` folder, for example: `(c:\ican50\edesigner\bin)`.

The Enterprise Designer Login dialog box appears.

**Figure 5** Enterprise Designer Login



- 2 Type your **Username** and **Password**. Click **Login** to start the Enterprise Designer.

**Note:** *If your login window doesn't appear or if you get an invalid Username or Password error, make sure your repository is running. Also, be aware that the Username and Password are case sensitive.*

### Create Project

A Project is a collection of logical components, configurations, and files that are used to solve a business problem.


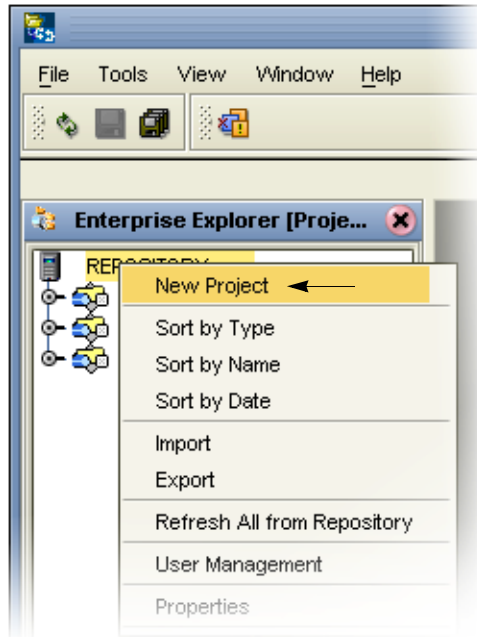
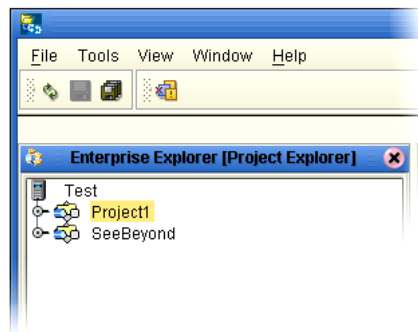
- 1 In the Enterprise Explorer pane of the Enterprise Designer, right-click the Repository name (computer icon ). Click **New Project**.

Figure 6 Create a Project



- 2 Accept the default **Project1** as the file name and press **Enter**.
- 3 The **Project1** structure appears in the Explorer pane on the left side of the window.

Figure 7 Project Folder: Project1



---

## 3.6 Create Connectivity Map

**Project1** needs a Connectivity Map to define the flow between the TimeCard system and the work file output.

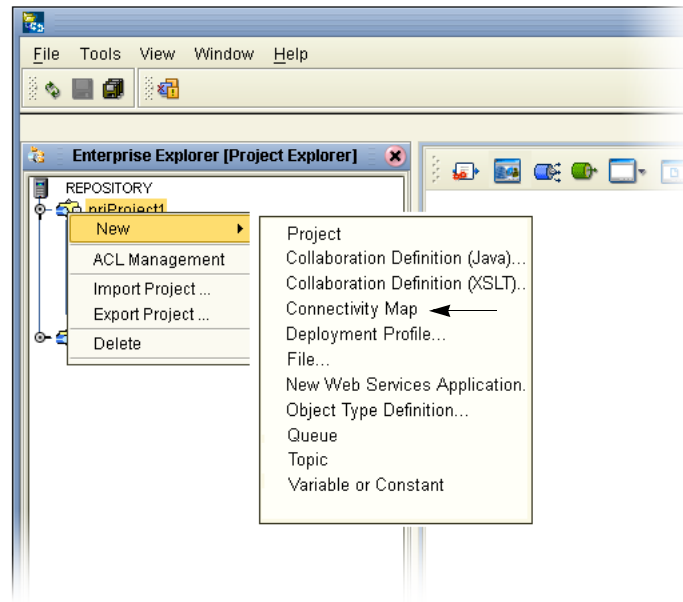
This section describes how to create the Connectivity Map for **Project1**.

## Name Connectivity Map

A Connectivity Map contains business logic and routing information about the data transmission. The Connectivity Map is a GUI (graphical representation) of the project.

- 1 In the Enterprise Explorer, right-click the **Project1** icon. Click **New** and **Connectivity Map**.

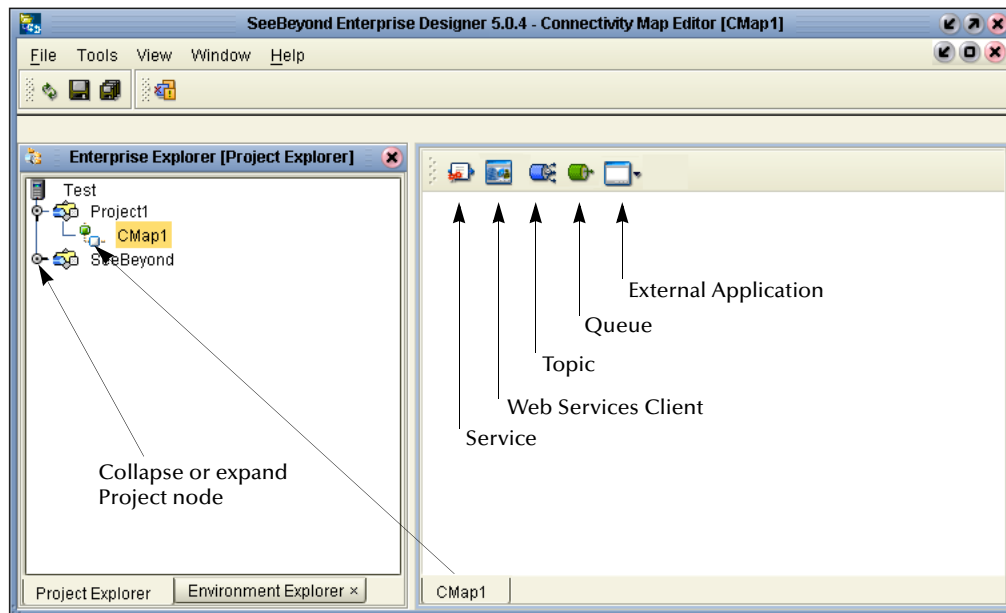
**Figure 8** Create a Connectivity Map



- 2 You may give your Connectivity Map a name. In this tutorial, accept the default name, **CMap1**.

**CMap1** is added to your **Project1**, as shown in the following figure.

**Figure 9** Project1 with a Connectivity Map (CMap1)



*Note:* The **CMap1** Connectivity Map appears as an icon in the Project Explorer. A **CMap1** tab is also added at the bottom of the Connectivity Map.

### 3.6.1 Add Components to the Connectivity Map

In this section, add the components to the Connectivity Map. Later link the components as shown in [Figure 3 on page 13](#).

In this section, add the following components to the Connectivity Map:

- **File1** (File External Application)
- **Service1**
- **File2** (File External Application)

#### Populate Connectivity Map

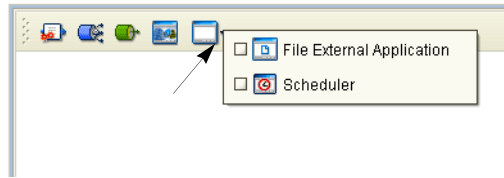
Place an input file (External Application) in your Connectivity Map.

- 1 On the Connectivity Map toolbar, click the External Application arrow (shown in [Figure 10](#)) to display the list.
- 2 Click **File External Application**.

*Note:* If the file option doesn't display in the drop-down list, you may need to install *FileWay.sar*.

This adds a new **File** icon to the toolbar.

**Figure 10** External Application Selection



- 3 Drag the **File** icon from the toolbar into the left side of the Connectivity Map.  
This creates a new File External Application called **File1**.
- 4 Click **Save** (CTRL+S).

*Note:* In Scenario2 you will name the components. In this scenario accept the default names.

### Create Service1

Create a Service to contain binding information about connecting the input to the output.

- 1 On the Connectivity Map toolbar, drag the **Service** icon into the Connectivity Map to the right of the **File1** icon.  
This creates a new Service called **Service1**.
- 2 Click **Save** (CTRL+S).

*Note:* To reposition a component in the Connectivity Map, click the icon and drag to a new location while holding down the mouse button.

### Create Output

Place an output file (External Application) in your Connectivity Map.

- 1 Drag another **File** icon into the Connectivity Map, and place it to the right of the **Service1** icon.  
This creates a new External Application file.
- 2 Click **Save** (CTRL+S).

**Figure 11** Connectivity Map



*Note:* Link these objects later.

## 3.7 Create New Object Type Definition

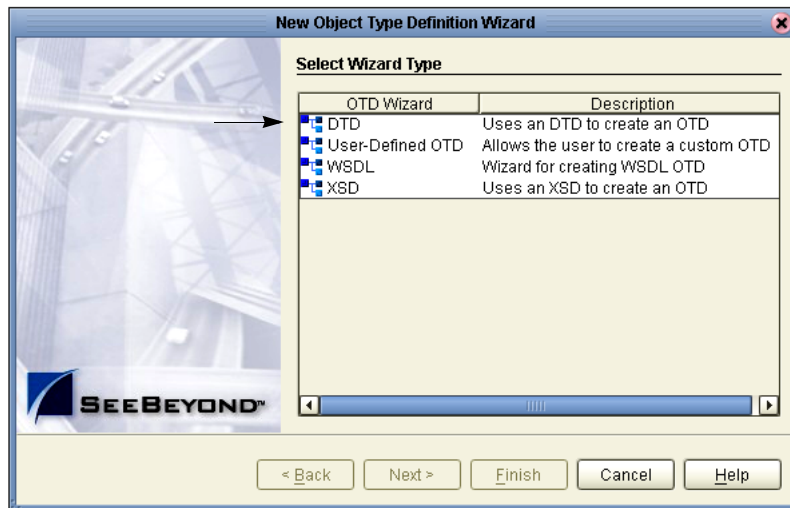
You must create an object type definition (OTD) before you can configure your Services.

### Create Object Type Definition

An OTD contains the structure and rules that define an object.

- 1 Right-click **Project1**.
- 2 Click **New** and **Object Type Definition**.

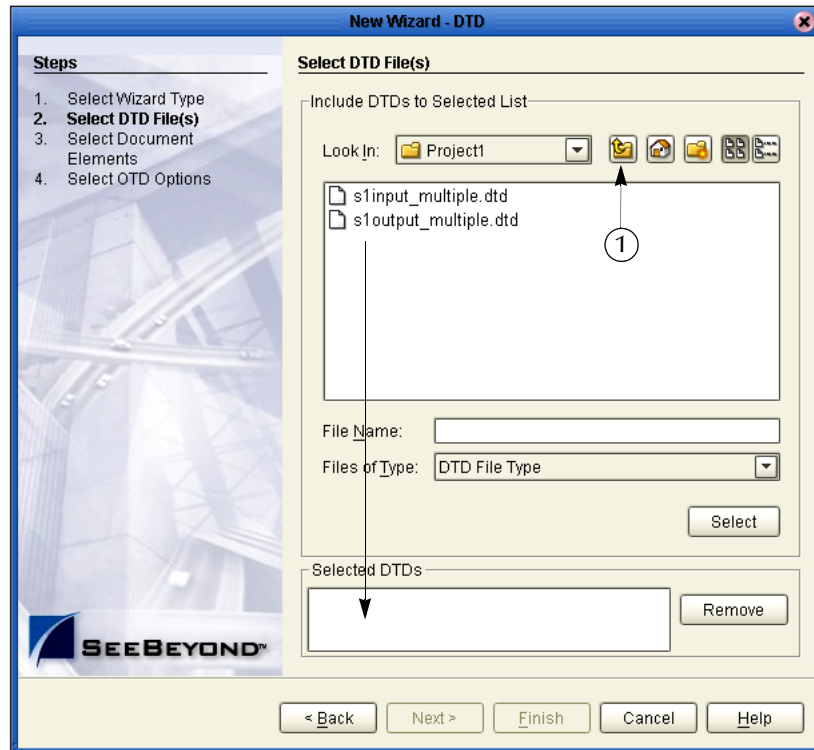
**Figure 12** OTD Wizard Selection



- 3 Select the **DTD** item.
- 4 Click **Next**.
- 5 Navigate to the sample data folder. For example, **c:\eGateData\Project1**.



Figure 13 Select DTD Files

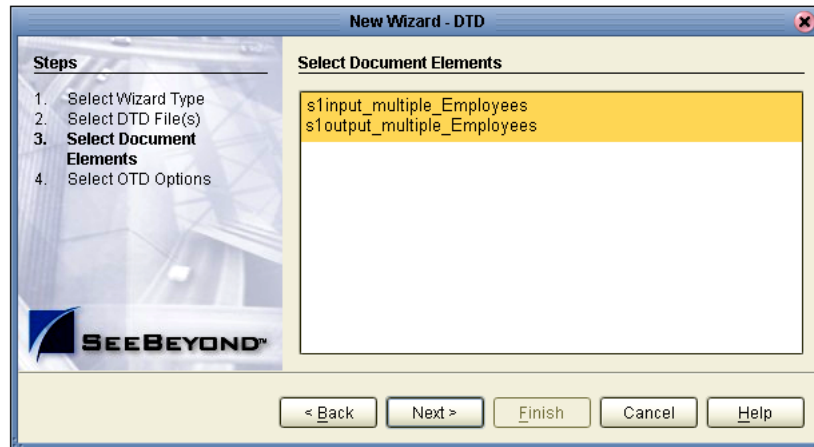


- 6 Highlight **s1input\_multiple.dtd**. Click **Select** (or double-click). The **s1** in the name stands for Scenario1.
- 7 Highlight **s1output\_multiple.dtd**. Click **Select** (or double-click).  
The **s1input\_multiple.dtd** and **s1output\_multiple.dtd** should appear in the Selected DTDs.
- 8 When both **s1input\_multiple.dtd** and **s1output\_multiple.dtd** appear in the Selected DTDs, click **Next**.

**Note:** The **Up one Level** icon (number 1 in Figure 13 above) is used to navigate to a previous screen; this is useful if you have made an incorrect selection.

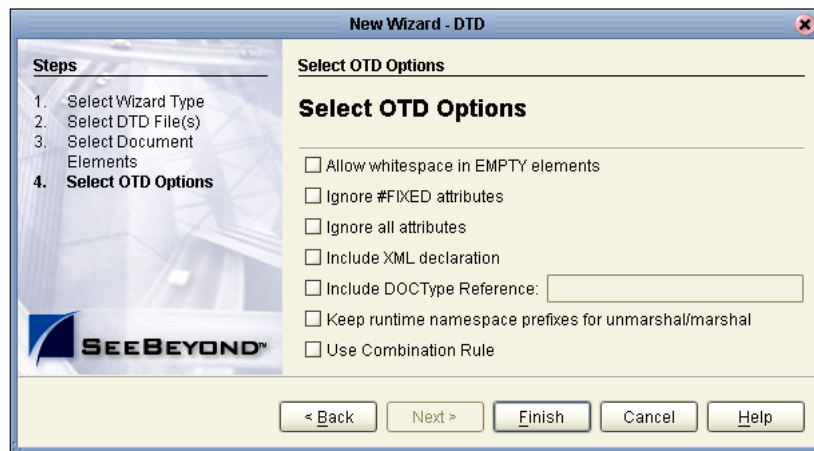
**Note:** As you progress through the steps, notice that the highlighted text in the left side of the New Wizard DTD window changes. In the previous step **Select DTD File(s)** was bolded. Now **Select Document Elements** is bolded, See Figure 14.

Figure 14 Select Document Elements



- 9 Select both `s1input_Employees_multiple` and `s1output_Employee_multiple` (hold down the CTRL key while selecting).
- 10 Click Next.

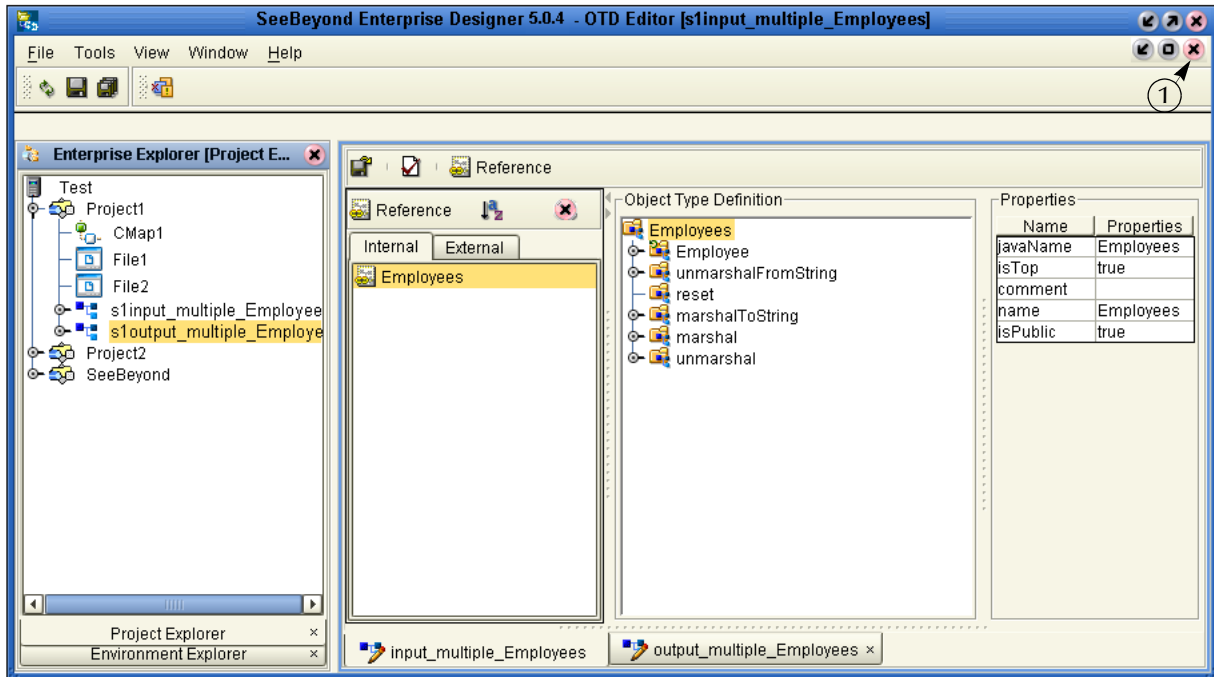
Figure 15 Select OTD Options



- 11 Click **Finish** (accepting the defaults).  
The OTD Editor window opens. See the following Figure 16.
- 12 On the **File** menu, click **Save**.

**Note:** *Save* saves all your changes in your current open editor while *Save All* saves all changes across all editors, including changes in the environments.

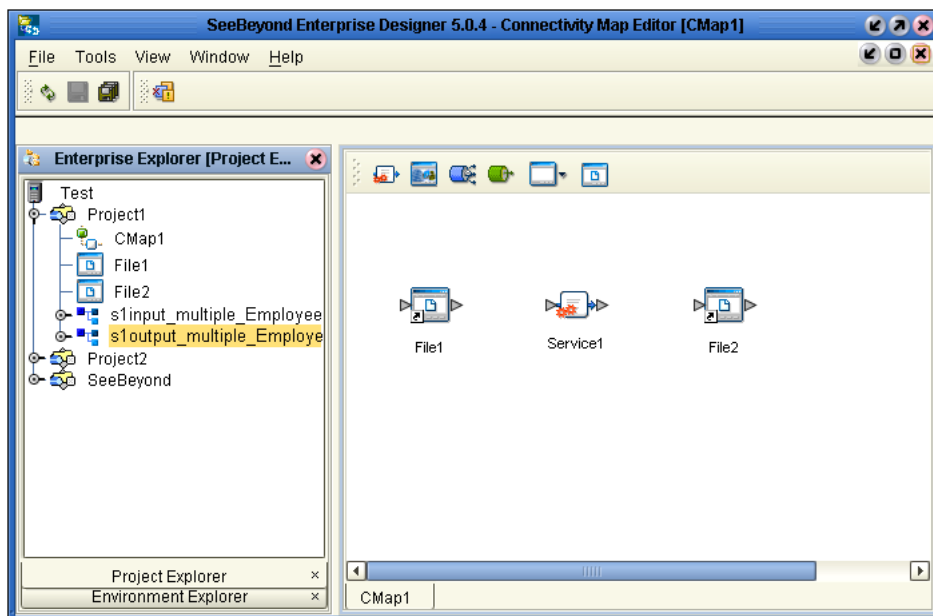
Figure 16 OTD Editor Input



- 13 Click the close window icon (number 1 in Figure 16) to redisplay the Connectivity map. (Or you may double-click the **CMap1** icon in the Project Explorer pane.)

The Connectivity map window appears. See the following figure.

Figure 17 Connectivity Map Window



## 3.8 Configure the Services

Use Service wizards in the following steps to configure your Collaboration.

### 3.8.1 Create Service1

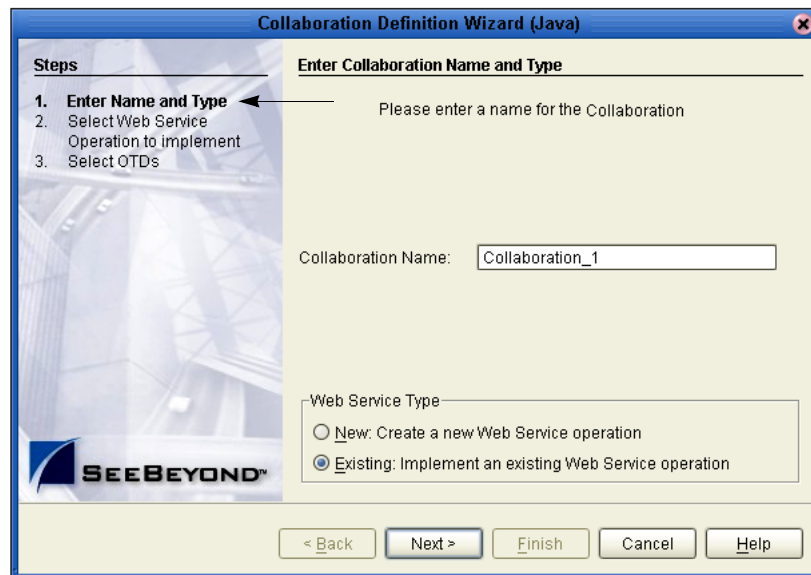
Configure a Collaboration (Java) definition.

#### Enter a Collaboration Name

- 1 Right-click **Project1**.
- 2 Click **New** and **Collaboration Definition (Java)**.

The Collaboration Definition Wizard (Java) appears.

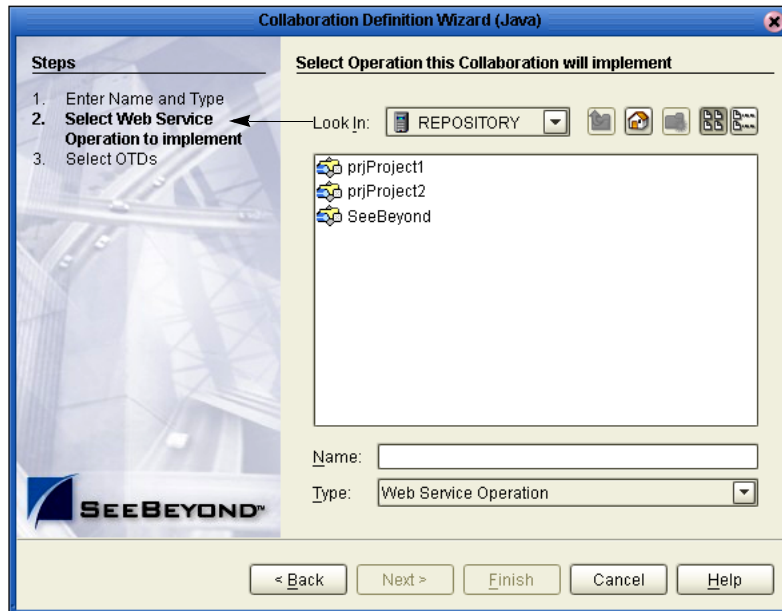
**Figure 18** New Collaboration Definition (Java) Name



**Note:** Accept the default name and the default radio button, **Existing Web Service**, for this Tutorial.

- 3 Accept the default name for the Collaboration Definition, **Collaboration\_1**.
- 4 Click **Next**.

**Figure 19** New Collaboration Definition Wizard (Java)



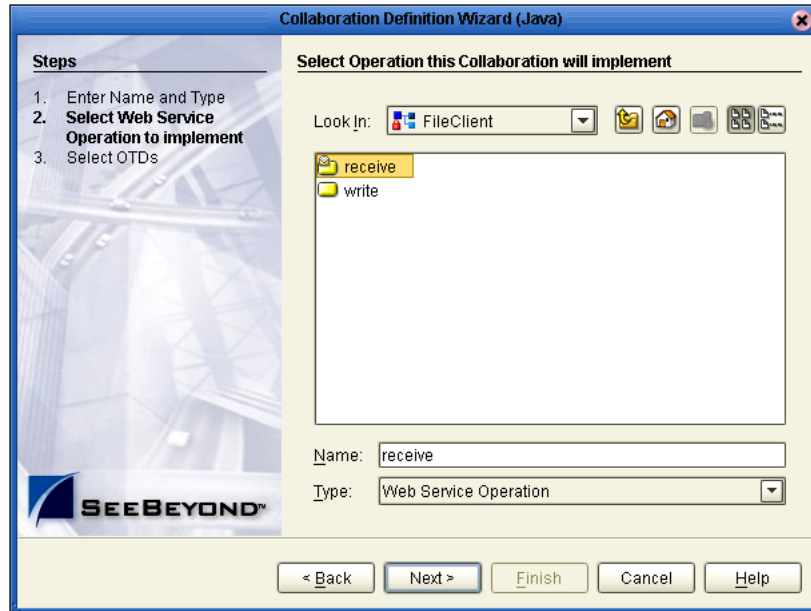
### Create Web Service Interface

Web Services enable communication between diverse applications using the Internet. Refer to *eGate Integrator User's Guide*, Chapter 6 for information about Web Services. Select a Web Service to “implement” a file-read.

- 1 Double-click SeeBeyond.
- 2 Double-click **eWays** (eWays are message end points that connect to an external service provider (system, person, or organization) to enable the sending and receiving of messages.)
- 3 Double-click **File**.
- 4 Double-click **FileClient**.
- 5 Click **receive**. Notice that **receive** appears in the Name field as shown in the following Figure 20.

(**FileClient.receive** is used for file-based eWays to bring data in or send data out.)

Figure 20 New Collaboration Web Service Interface

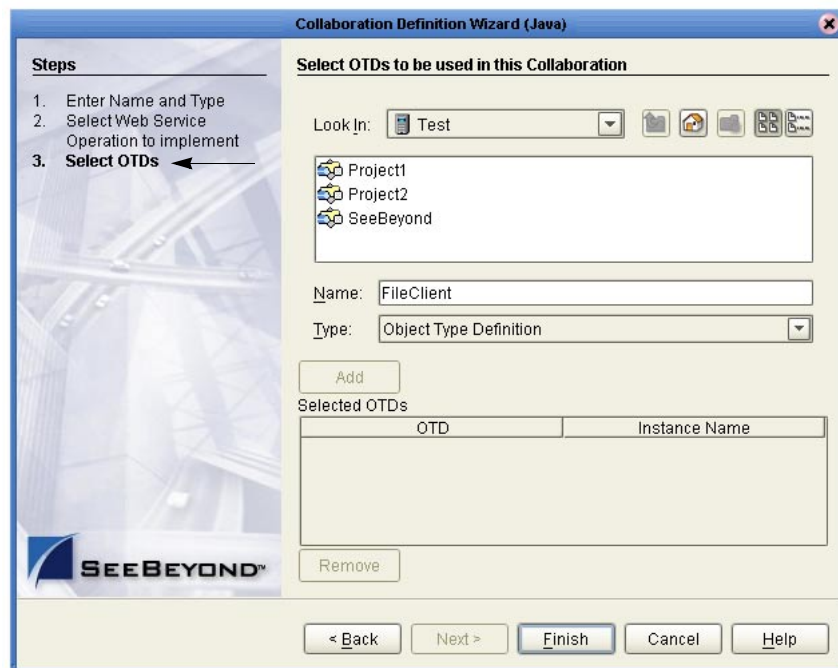


6 Click **Next** when “receive” appears in the File Name field.

#### Select OTDs

In the following steps select the **FileClient.receive** OTD (its parameters are automatically defined). Also, select the user defined OTDs.

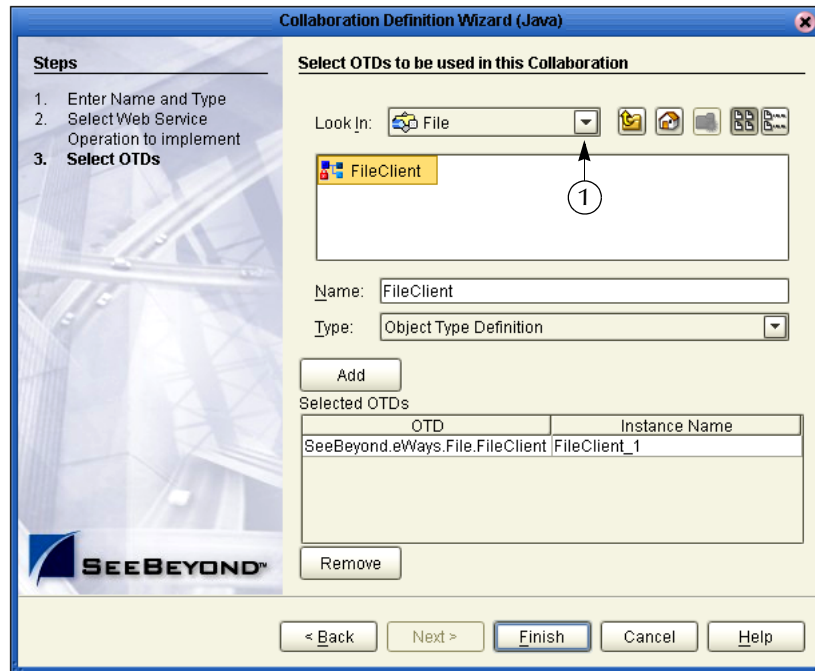
Figure 21 Select OTDs



- 1 Double-click **SeeBeyond**.
- 2 Double-click **eWays**.
- 3 Double-click **File**.
- 4 Double-click **FileClient**.

This adds **FileClient\_1** to the list. See the following Figure 22.

**Figure 22** Select FileClient OTD

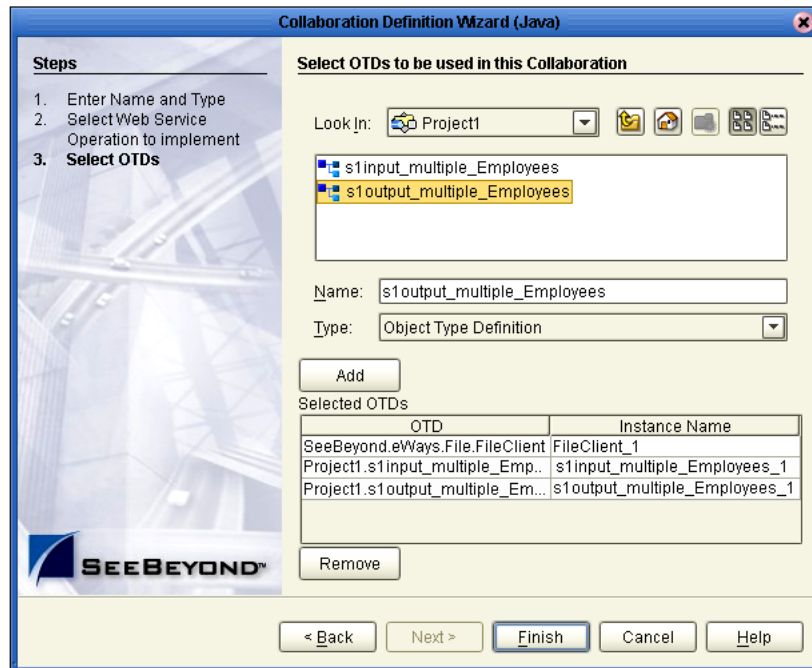


Don't Click Finish yet. You need to install the user defined OTDs.

- 5 Click the drop-down arrow (number 1 in Figure 22).
- 6 Click **Project1** to display the file OTDs.
- 7 Double-click **s1input\_multiple\_Employees**.
- 8 Double-click **s1output\_multiple\_Employees**.

See the following figure.

Figure 23 Selected OTDs



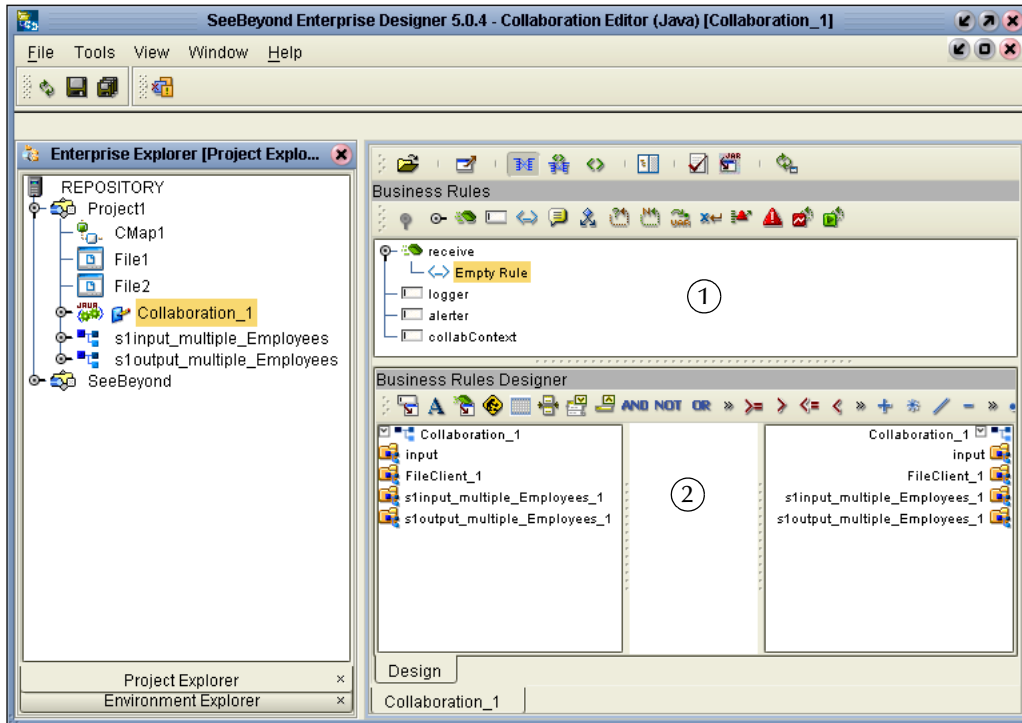
All three OTDs appear in the list of OTDs.

9 Click **Finish**.

The Collaboration Editor (Java) appears. See the following figure.



Figure 24 Collaboration Editor (Java)



## 3.9 Apply the Business Rules

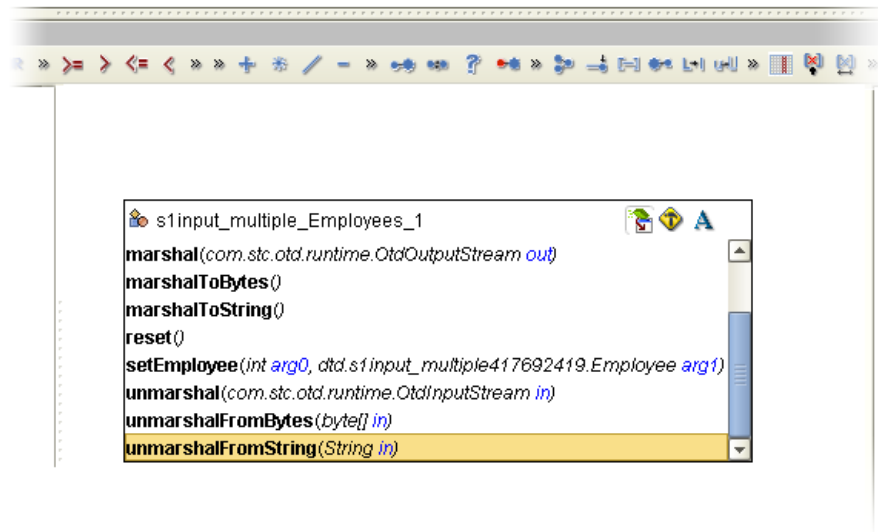
Business Rules display in the Business Rules pane (number 1 in Figure 24). Use the Business Rules Designer GUI (number 2 in Figure 24) to map your rules.

### Unmarshal Text

You must first unmarshal the text input to prepare for a “read” before you can concatenate the elements.

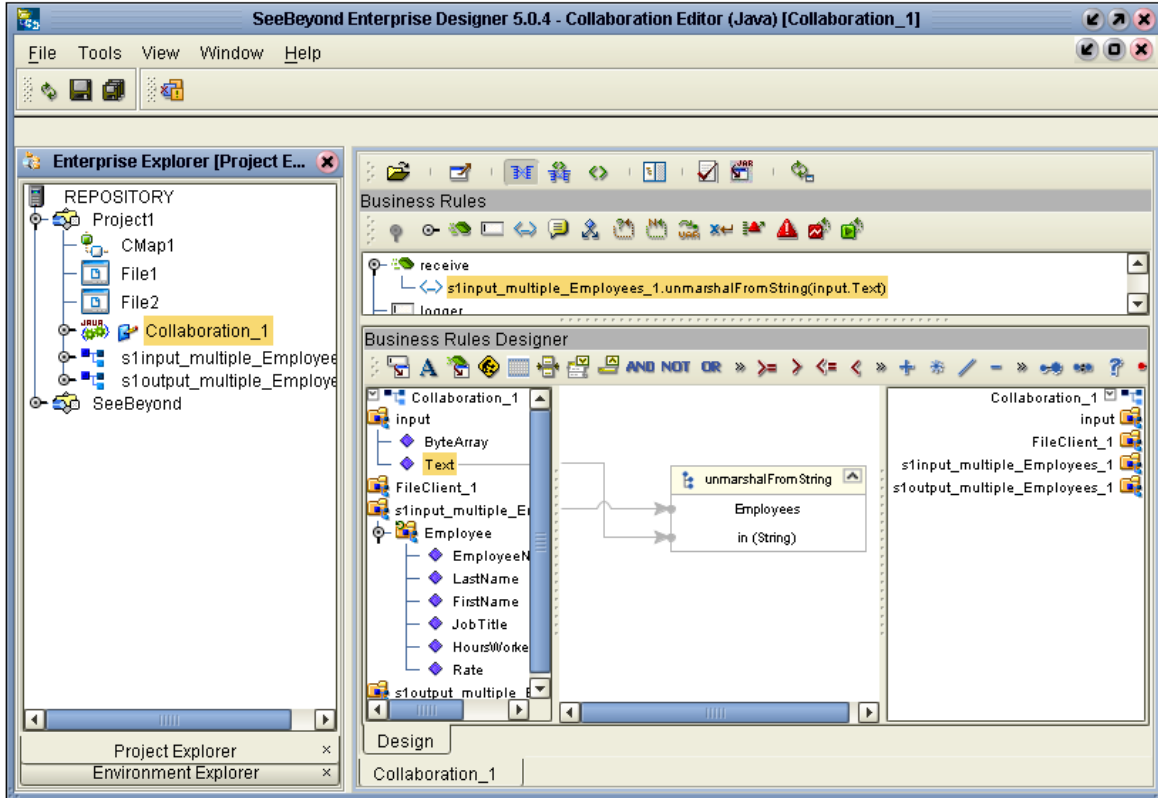
- 1 Double-click **input** in the left pane of the Business Rules Designer to expand the view.
- 2 Double-click **s1input\_multiple\_Employees\_1** in the left pane of the Business Rules Designer to expand the view (and double-click **Employee**).
- 3 Click and drag **s1input\_multiple\_Employees\_1** to the Business Rules Designer pane.

Figure 25 Select Unmarshal



- 4 Double-click **unmarshalFromString(String in)** in the list.  
This action places a method box into the center pane of the Business Rules Designer canvas. The unmarshal method reads your employee input XML file.
- 5 Drag Text from **input** into **in(String)**. See the following figure.

Figure 26 Unmarshal Text



### Connect Input and Output OTD Nodes

In the following steps connect **FirstName** and **LastName** in **s1input\_multiple\_Employees\_1** in the left pane with **FullName** in **s1output\_multiple\_Employees\_1** in the right pane (using **concat**).


- 1 Double-click to expand the view of **s1output\_multiple\_Employees\_1** and **Employee** in the right pane.
- 2 Double-click to expand the view of **s1input\_multiple\_Employees\_1** and **Employee** in the left pane (if not already expanded).

**Note:** Notice that the objects in both the left and right pane of the Business Rules Designer window have the same name. Keep in mind that the GUIs (**FileClient\_1**, **s1input\_multiple\_Employee\_1**, and **s1output\_multiple\_Employee\_1** for example) refer to the same object whether they are in the left or right pane.

- 3 Connect **EmployeeNumber**, from **s1input\_multiple\_Employees\_1** in the left pane to **EmpNo** of **s1output\_multiple\_Employees\_1** in the right pane.

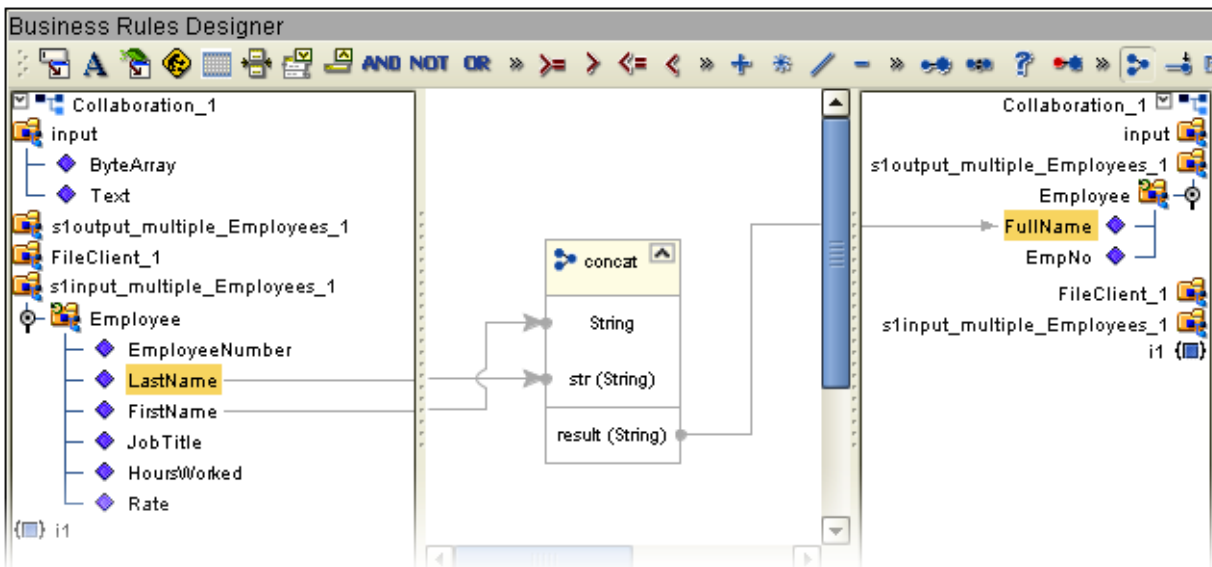
This action connects the nodes and also sets up the loop logic to process multiple records.

**Note:** As you begin mapping, the graphic (GUI) disappears. Click any business rule in the top pane to redisplay the business rule.

- 4 Drag the **Concat** icon  down to your designer workspace.
- 5 Connect **FirstName** to **String** in the Concat box. (Connect from **s1input\_multiple\_Employee\_1** in the left pane.)
- 6 Connect **LastName** to **str (String)** in the Concat box. (Connect from **s1input\_multiple\_Employee\_1** in the left pane.)
- 7 Connect **result (String)** to **FullName** in **Employee** of the root element **s1output\_multiple\_employees\_1** in the right pane.

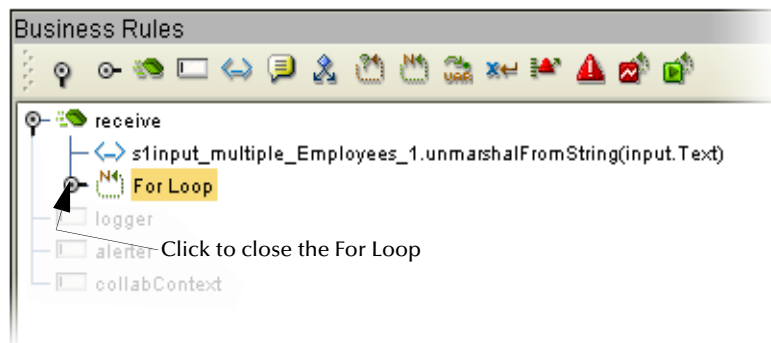
Do not select a method at this time. See the following figure.

Figure 27 Concatenation Logic



- 8 Click the "For Loop" node expansion control to collapse the loop. See the following figure.

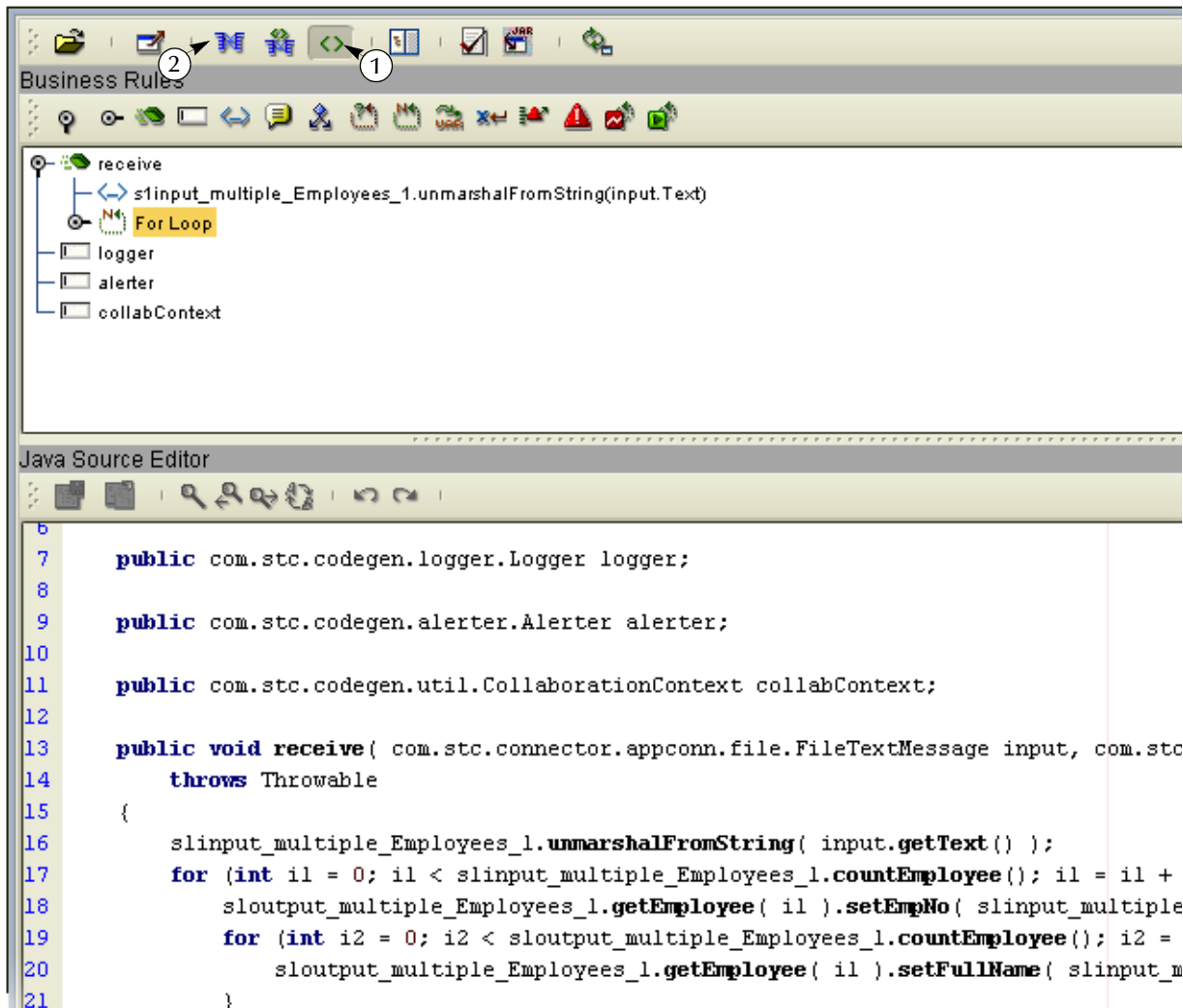
Figure 28 Collapse For Loop



## View Source Code

You may view or modify the source code by clicking the **Source code mode** icon (number 1 in Figure 29). You may toggle back to Standard mode by clicking the **Standard mode** icon (number 2 in Figure 29).

Figure 29 Source Code Mode



## Marshal Text

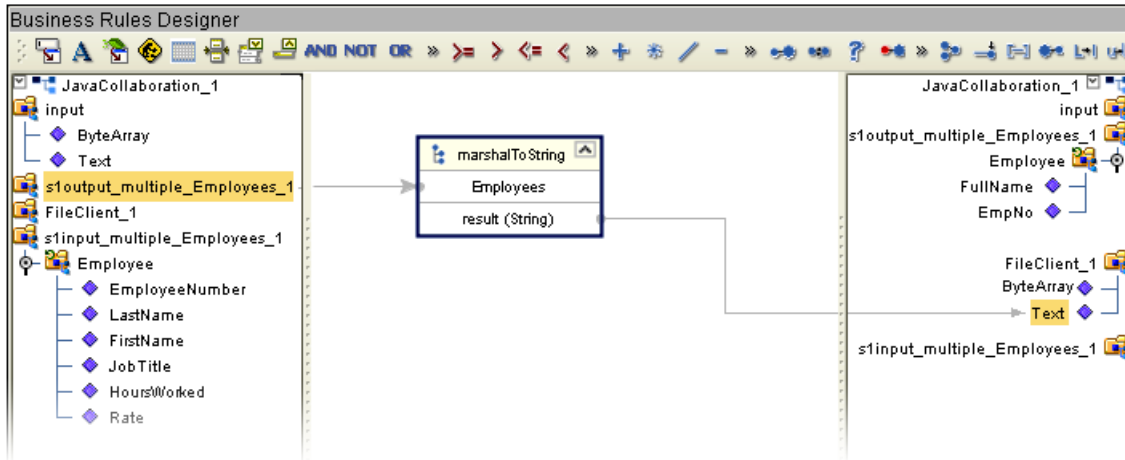
Since you began by unmarshalling data from string to prepare to read, you must now marshal the data to string to prepare to write.

- 1 Double-click **FileClient\_1** in the right pane to expand the view.
- 2 Click and drag **s1output\_multiple\_Employees\_1** from the left pane to the Business Rules Designer pane.
- 3 Select and double-click **marshalToString()**.

The **marshalToString** method box appears.

- 4 Connect a node from **result (String)** to **Text** under **FileClient\_1** in the right pane. Do not select a method at this time. See the following figure.

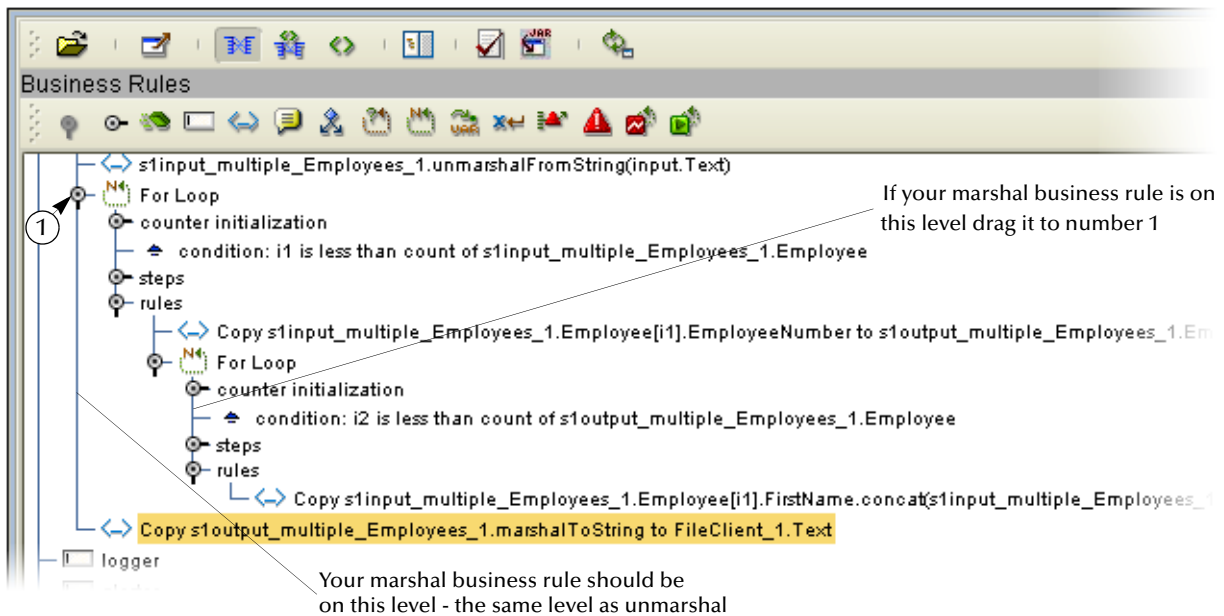
**Figure 30** Marshal to String



Your Business Rules (in the top pane) should look like Figure 31. The marshal and unmarshal actions should be on the same level. The “marshal” and “write” (which you will add next) should not be within the loop because that would cause a write after each individual employee record.

**Note:** If your “marshal” rule is on the same level as the concat rule, click and drag your “marshal” outside the loop to number 1 in Figure 31.

**Figure 31** View Business Rules





## 3.10 Apply the Collaborations

You have configured your Collaboration Definition (business rules) in the previous section. You are now ready to bind your Collaboration Definition with the Service using the drag-and-drop method.

### 3.10.1 Link Components in the Connectivity Map

Linking the components creates the logical flow of data throughout the Project. Additionally, creating a link between the files and the Service adds eWays to the Connectivity Map.

At this point, the Connectivity Map contains three unlinked objects.

**Figure 33** Objects in Connectivity Map

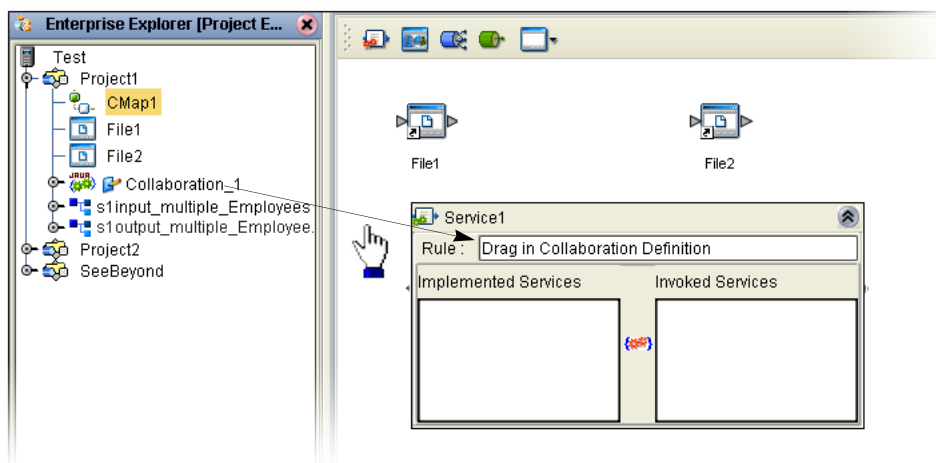


#### Connect Nodes

- 1 Double-click **Service1** in the Connectivity Map to open the Service window. (You can also reposition the window after it is open.)
- 2 Drag the **Collaboration\_1** definition from the Project Explorer pane to the **Rule** field in the open window as shown in the following Figure 34.

**Note:** Click once on **Collaboration\_1** to activate Project Explorer in the left pane. Then click **Collaboration\_1** once again to select and drag. You may also drag **Collaboration\_1** into the icon without first opening the Collaboration window.

**Figure 34** Using the Drag and Drop method



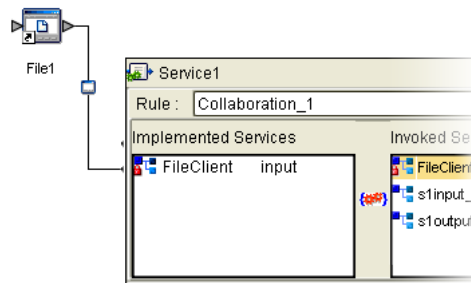


Notice that the gear icons change from red to green to signify that the Collaboration is bound to the Service.

**Note:** You may connect nodes forward or backward. However, it may be easier to connect from the Service dialog box backward to the module icon, especially when there are multiple Services in more complex projects.

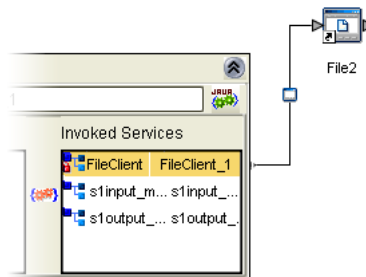
- 3 Connect **FileClient** input to **File1** as shown in the following Figure 35. Since there is only one option the application automatically makes the connection.


**Figure 35** Connect FileClient to File1



- 4 Connect **FileClient** to **File2** as shown in the following Figure 36.

**Figure 36** Connect FileClient to File2



- 5 Click the **Close** button  to close the binding box.
- 6 Click **Save All**.

---

## 3.11 Configure the eWays

Your Connectivity Map contains two eWays. These must be configured for deployment.

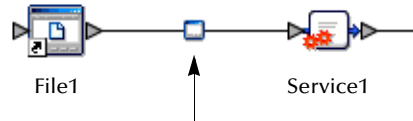
- The input eWay polls a directory on the local file system. For example: **c:\eGateData\Project1**, for any XML file (\*.xml).
- The Output eWay writes the results to an output file (.dat) on the local file system. For example: **c:\eGateData\Project1**.

### Configure Inbound eWay

Identify the file eWay, shown in the following Figure 37, as an inbound file and then set the properties to identify the file and its location.

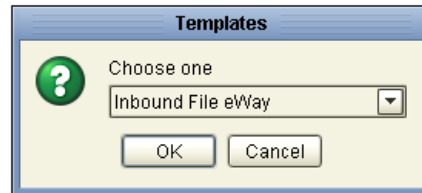
- 1 Double-click the first eWay . Select **Inbound File eWay** from the drop-down list.

Figure 37 Inbound eWay



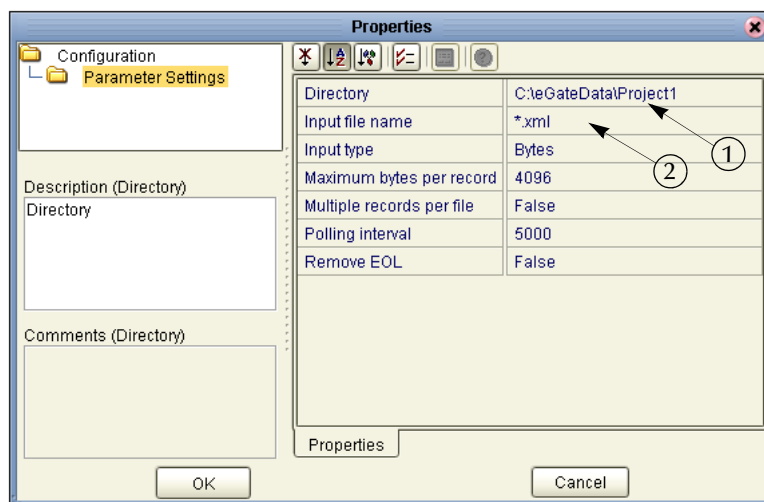
- 2 Click **OK**.

Figure 38 External Type Dialog Box



This creates a new inbound eWay and the Properties Dialog Box appears.

Figure 39 Properties Configuration (Inbound eWay)



Refer to the arrows in the previous figure. These are the editable fields you configure.

- 3 Click once on the current path to select and type a correct directory path. In this tutorial the path is **c:\eGateData\Project1** (number 1 in Figure 39 above).

- 4 Click once on the current **Input File Name** field and enter \*.xml. The eWay reads any files with the .xml extension (number 2 in Figure 39 above).
- 5 Click **OK** to close the Properties Dialog Box for the inbound eWay.

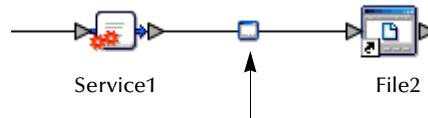
**Note:** If the **Polling interval** value is left unchanged, the eWay polls the directory location every five seconds (5000 milliseconds).

### Configure Outbound eWay

Identify the file eWay shown in the following Figure 40 as an outbound file and then set the properties to identify the file and its location.

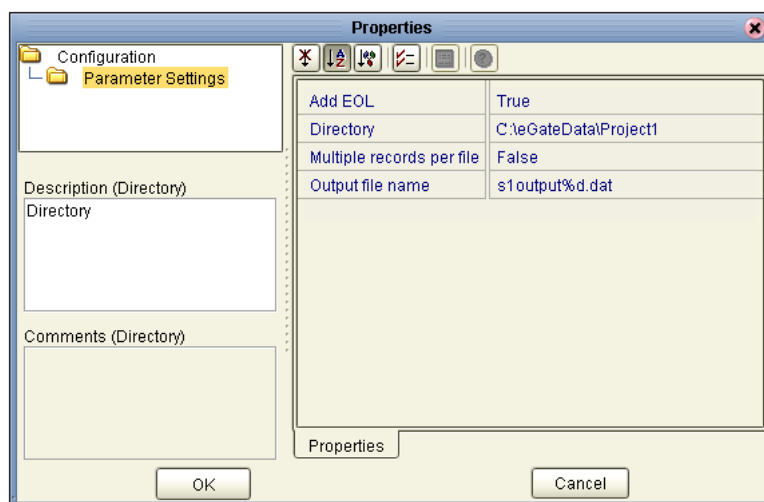
- 1 Double-click the second eWay.
- 2 Select **Outbound File eWay** from the drop down list.

**Figure 40** Outbound eWay



- 3 Click **OK**.  
The Properties dialog box appears.  
Use the same path you used for the inbound eWay (c:\eGateData\Project1).
- 4 Name the output file: **s1output%d.dat** (Scenario1 output).
- 5 Click **OK** to close the Properties Dialog Box for the outbound eWay.

**Figure 41** Properties Configuration (Outbound eWay)



**Note:** The *Multiple records per file* setting can be confusing. You may use either *True* or *False* in the tutorials. Refer to “**Multiple records per file (True or False)**” on [page 44](#).

6 On the **File** menu, click **Save All**.

You have now configured both the input and output eWays and are ready to deploy your Project. You may skip to “**Create Environment**” on [page 44](#).

## Multiple records per file (True or False)

This setting (refer to Figure 39 and Figure 41) specifies if multiple records are to be obtained or sent per file. Multiple records (messages) are generated per line up to the maximum bytes per record.

### Inbound eWay

A **True** setting means that multiple messages are generated per line up to the number specified in the Maximum bytes per record property.

A **False** setting, as in the tutorial, means that the XML file is read in as a block and represents a single message. This message can be treated as one “record” because our business rule contains the logic to loop through each record in the XML message.

### Outbound eWay

For the output eWay, this setting determines whether records are concatenated in a file.

A **True** setting means that multiple messages are written to the same file. New messages are appended (concatenated) to the output file with each **File.send()**.

A **False** setting means that only one message is written in one file. New messages are written to a different file (with the file number incremented if you use %d) with each **File.send()**.

You may want to experiment with this setting and observe the different results.

---


## 3.12 Create Environment

Create and activate the Deployment Profile. Before you can create the profile, you must create an Environment.

### 3.12.1 Create Environment and Logical Host

An Environment is a collection of physical resources and their configurations that are used to host Project components. An Environment contains logical hosts and external systems.

1 Click **View** from the Menu bar. Click **Environment Explorer** (or click the **Environment Explorer** tab).

- 2 Right-click the Repository name (  ). Click **New Environment**.
- 3 Right-click **Environment1** and rename **Tutorial**. Press **Enter**.
- 4 Right-click **Tutorial**. Click **New Logical Host**.

This creates a **LogicalHost1** window in the right pane.

### Create Integration Server

An integration server is a J2EE software platform that houses the business logic container used to run Collaborations and JCA connectors (eWays).

- 1 Right-click **LogicalHost1** in the **Enterprise Explorer** window. Click **New SeeBeyond Integration Server**.

**IntegrationSvr1** appears in the **LogicalHost1** window. Your Collaboration is bound to this server.

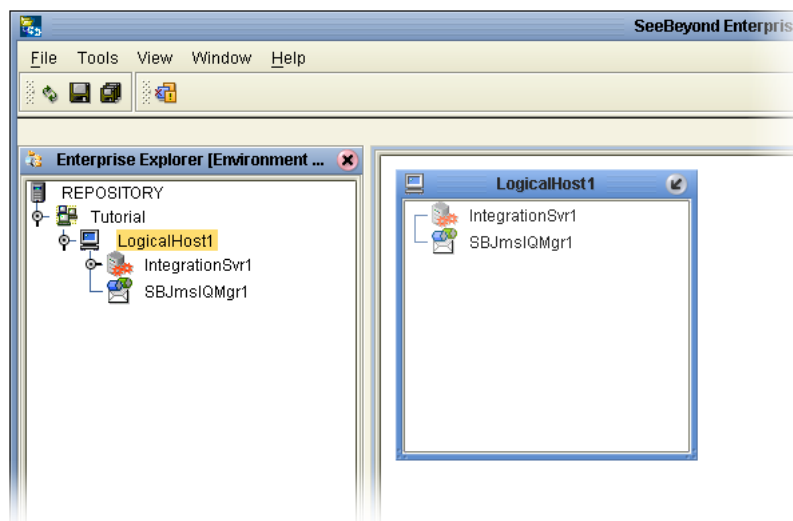
### Create JMS IQ Manager

The JMS IQ Manager is a JMS-compliant, guaranteed delivery store, forwarding, and queueing service. (You would bind your Topics and Queues to this server if you had them. In Scenario1 there are no Topics or Queues; in Scenario2 you will use a Topic.)

- 1 Right-click **LogicalHost1** in the **Enterprise Explorer** window. Click **New SeeBeyond JMS IQ Manager**.

**SBJmsIQMgr1** appears in the **LogicalHost1** window.

**Figure 42** Environment with Logicalhost



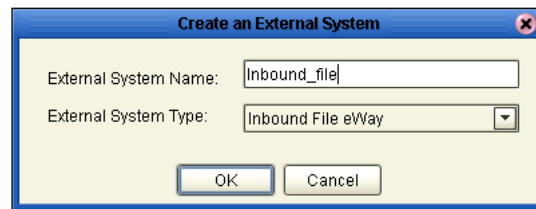
These servers automatically appear in the **LogicalHost1** dialog box.

### Create External File System for Inbound eWay

- 1 Right-click **Tutorial** (Environment).

- 2 Click **New File External System**.

**Figure 43** External File Inbound



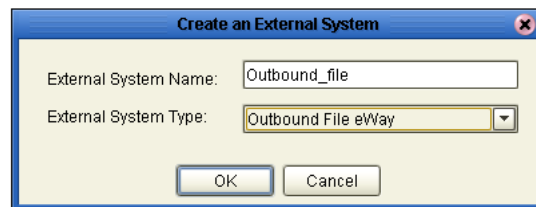
- 3 Select **Inbound File eWay**.
- 4 Type **Inbound\_file** in the name field.
- 5 Click **OK**.

This places the inbound external system, **Inbound\_file** in the Environment Editor pane.

#### Create External File System for Outbound eWay

- 1 Right-click **Tutorial**.
- 2 Click **New File External System**.

**Figure 44** External File Outbound



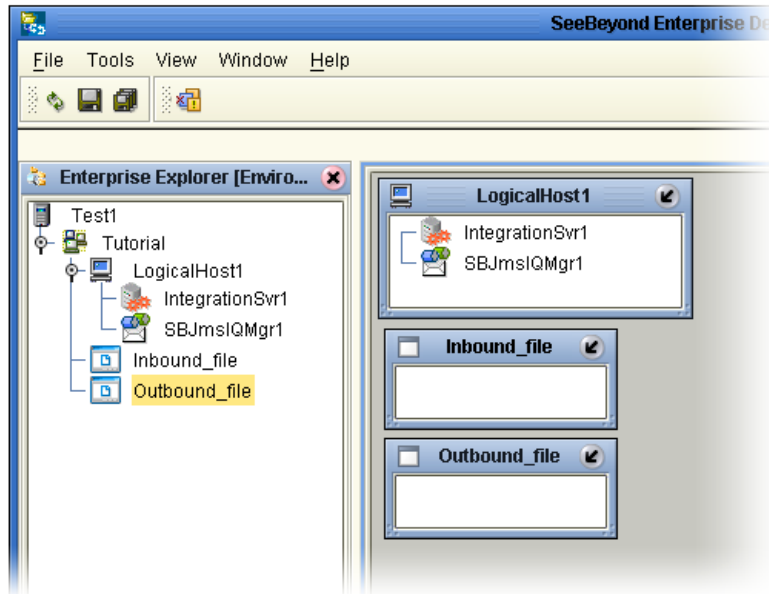
- 3 Select **Outbound File eWay**.
- 4 Type **Outbound\_file** in the name field.
- 5 Click **OK**.

This places the outbound external system, **Outbound\_file** in the Environment Editor pane.

**Note:** If the "Create an External System" (or any Java pop-up) window doesn't appear, it may be hidden behind your Enterprise Explorer window. Press the ALT+TAB keys to see if you have any hidden pop-ups. Use this keystroke combination if your Enterprise Explorer GUI appears to be "frozen."

Your Environment Editor pane should look like the following figure.

Figure 45 Environment Editor



---

### 3.13 Create and Activate Deployment Profile

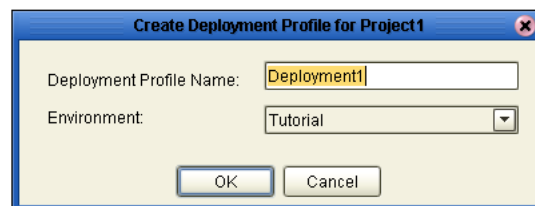
First create a Deployment Profile, and then deploy (map) the components.

#### Create Deployment Profile

A Deployment Profile contains information about how Project components are deployed in an Environment, and it also maps components to the Environment.

- 1 Click the **Project Explorer** tab to return to the Project Explorer pane of your Project.
- 2 Right-click **Project1**, and then click **New** and **Deployment Profile**.

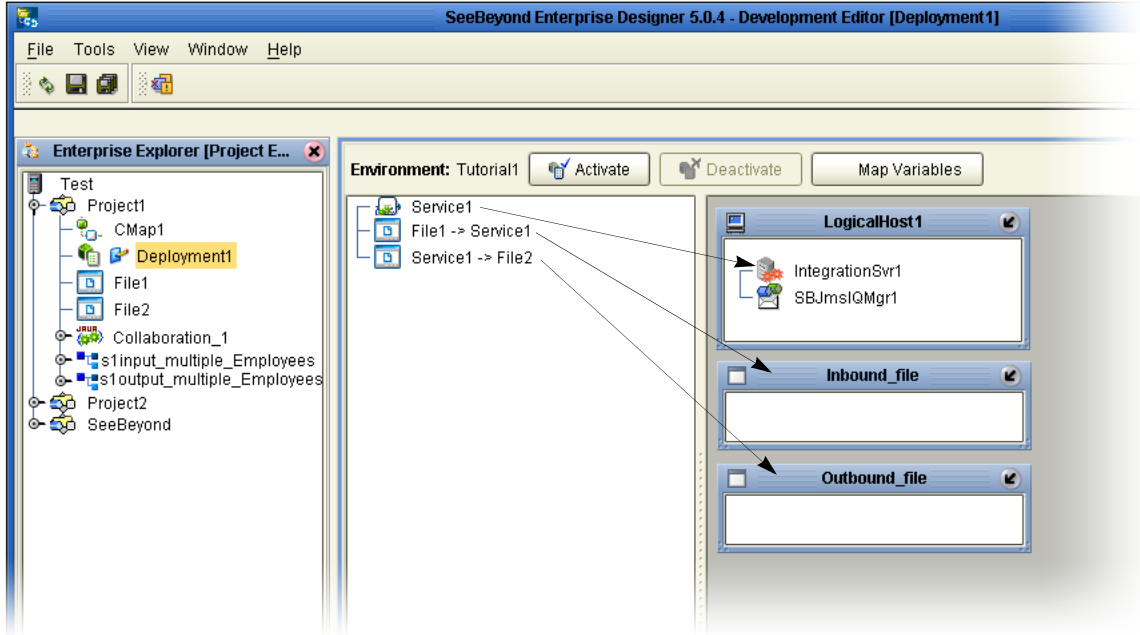
Figure 46 Deployment Profile



- 3 Accept the default name **Deployment1** (and **Tutorial**). Click **OK**.

Your files and Service1 appear in the center pane of the Environment editor. Deploy these components using the drag and drop method.

Figure 47 Environment, Drag and Drop Components



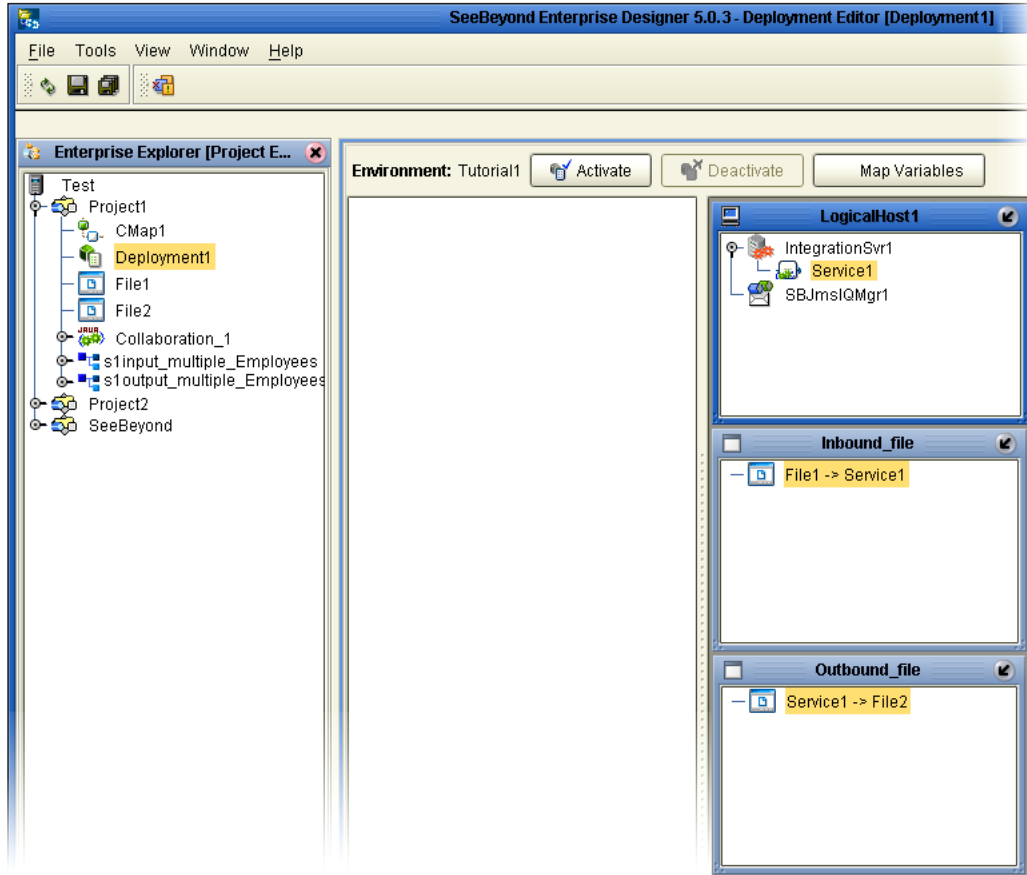
### 3.13.1 Deploy Project1

Drag and drop components into the server dialog boxes.

- 1 Drag-and-drop your Service, **Service1**, into your Integration Server, **IntegrationSvr1**. (See the previous Figure 47.)
- 2 Drag-and-drop your external files into your External File Servers. (File1 is your inbound file and File2 is your outbound file.)



Figure 48 Populated Environment



After populating your Environment with all the available components, the center pane of the Environment editor is empty. All your components reside in the servers, as shown in the previous Figure 48.

---

## 3.14 Run the Bootstrap and Management Agent

The Bootstrap command starts the Logical Host and begins the process of polling your input data. The bootstrap process is run from a command prompt or by executing **bootstrap.bat**. Bootstrap picks up the deployment profile the first time it runs; after that reactivate and check the box in Figure 51 to apply the most recent changes to the logical host(s).

### Logical Host Overview

At run time, the Logical Host bootstrap script starts the bootstrap Java program that downloads the Management Agent, the Message Server, and the Integration Server from the Repository. The Management Agent is then started, which in turn starts the Message Server(s) and Integration Server(s).

On subsequent bootstraps of the Logical Host, the configuration will not automatically update and you can continue running the previous version of the deployed components. You can force a refresh of the Logical Host configuration by using the **-f** option.

Enter the command: **bootstrap -f**

You can also refresh the Logical Host using the option on the Activate dialog box. See Figure 51.

---

## 3.15 Run the Bootstrap

*Note:* The Bootstrap command is case sensitive on Windows platforms.

Two ways to run the bootstrap are explained below:

### Edit Properties and run bootstrap.bat (method 1)

You can edit the configuration file: *Logical-host.properties*. Ensure that the Logical Host is not running. The required fields are explained below:

- 1 Navigate to the following file (depending your installation's setup):  
Ican50>logicalhost>bootstrap>config>**logical\_host.properties**.

These properties are automatically persisted by the bootstrap sequence. Default properties are used if none are provided at the command line.

- 2 Edit: **repository.url=http://localhost:12000/Test**  
localhost:12000 the default port number. Test is the repository name in this example.
- 3 Edit: **repository.username=Administrator**  
Enter the Administrator name.
- 4 Enter the repository password: **repository.password=**  
repository.password: (USER CONFIGURABLE). Plain text form of password used for connecting to the repository. Any value provided here are cleared out by the system and written in encrypted form to the **repository.password.encrypted** field.
- 5 Edit: **logical.host.environment.name=Tutorial**  
Enter an environment name that is used when you deploy your project.
- 6 Edit: **logical.host.name=LogicalHost1**  
logical.host.environment.name: (USER CONFIGURABLE). Specifies the name of the environment containing the current logical host.  
After you have configured this file you can execute **bootstrap.bat**.  
Example: C:\Ican50\logicalhost\bootstrap\bin\**bootstrap.bat**

*Note:* Do not enter spaces before or after the equal sign (=) and the property values. Spaces are allowed only in the value itself.

### Run Bootstrap.bat from a DOS command (method 2)

You can also enter the bootstrap commands in a **cmd** shell as explained in the following steps.

- 1 Open a Windows command prompt as shown in the following Figure 49. (Click **Start**; click **Run**; type **cmd**.)
- 2 Navigate to where you installed the logicalhost. For example, **C:\ican50\logicalhost\bootstrap\bin**
- 3 To start the bootstrap process, type the following command:

```
bootstrap -e environment_name -l logicalhost_name  
-r repository_URL -i username -p password
```

Where *environment\_name* is the name of your environment (for example, *Tutorial*), *logicalhost\_name* is the name of your Logical Host (for example, *LogicalHost1*), *repository\_URL* is the full URL of your Repository including the Repository name (for example, *http://labserver:12000/Test*), *username* is your user name, and *password* is your password.

**Figure 49** Bootstrap Command Example



- 4 Press **Enter**.

The bootstrap process takes a few minutes to execute. The Management Agent starts the components in the Project; the Input eWay processes the input XML message.

*Note:* To force a refresh of the Logical Host see [“Logical Host Overview” on page 49](#).

---

## 3.16 Run Project1

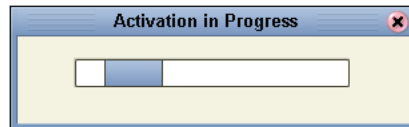
When you deploy **Project1**, a deployment file is created and later is picked up by the Logical Host. The Project you created with Enterprise Designer is “code generated” into a package that is passed to the Repository. The Logical Host picks up this package from the Repository.

If you want to do a “hot deploy,” which means that the Logical Host picks up the current package in real time after the bootstrap has been started, you could click **YES** when prompted. See the following Figure 51.

### Activate Environment

- 1 On the **File** menu, click **Save All**.
- 2 Click the **Activate** button. The Activation in Progress message appears.

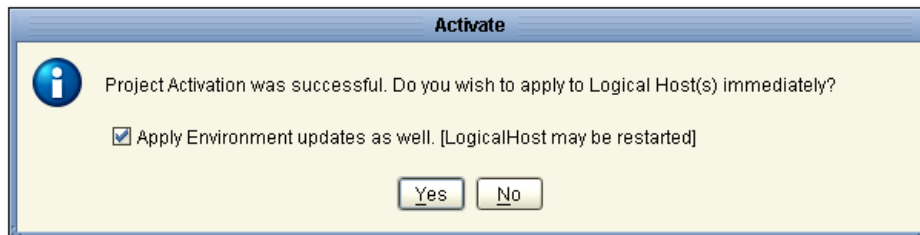
**Figure 50** Activation in Progress Message



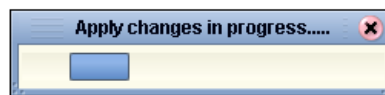
- 3 Check the box, *Apply Environment Updates*, the first time you execute the bootstrap, then click **OK**. See the following figure.

**Note:** *If you do not check the box, the Management Agent will not pick up the current Environment changes but will only pick up the runnable project file.*

**Figure 51** Activation Options

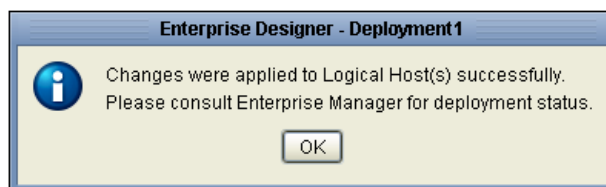


**Figure 52** Apply Changes in Progress



Applying changes to the Deployment Profile may take a few minutes. If your changes are successful the following message appears.

**Figure 53** Changes Applied to Logical Host



- 4 Click **OK**.

*Note:* You can make changes while the Project is running and then Reactivate to get the latest changes.

---

## 3.17 Verify Output Data

The Management Agent processes your input file, **TimeCard\_Multiple.xml**, and writes the results to **s1output1.dat**.

### View Output

- 1 Use Windows Explorer to navigate to **c:\eGateData\Project1**.
- 2 See that the Input eWay, after processing the input file, renamed the input file to **TimeCard\_Multiple.~in**.
- 3 See that the Output eWay generated the output file, **s1output1.dat**.
- 4 Use Windows Wordpad or a similar text edit application to view the contents of the output file.

**Figure 54** Output File

```
<?xml version="1.0" encoding="UTF-8"?>
<Employees>
  <Employee>
    <FullName>DekotaDouglas</FullName>
    <EmpNo>100</EmpNo>
  </Employee>
  <Employee>
    <FullName>LingChang</FullName>
    <EmpNo>123</EmpNo>
  </Employee>
  <Employee>
    <FullName>DickJamison</FullName>
    <EmpNo>118</EmpNo>
  </Employee>
  <Employee>
    <FullName>TakeshiNakayama</FullName>
    <EmpNo>144</EmpNo>
  </Employee>
</Employees>
```

In this sample, the first and last names are concatenated in the field, **FullName** as shown in Figure 54. (The test files are editable and you can experiment with the input. Your data may not match the samples shown in the Tutorial.)

Figure 55 Input File

```
<?xml version="1.0" encoding="UTF-8"?>
<Employees>
  <Employee>
    <EmployeeNumber>100</EmployeeNumber>
    <LastName>Douglas</LastName>
    <FirstName>Dekota</FirstName>
    <JobTitle>Manager</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>55</Rate>
  </Employee>
  <Employee>
    <EmployeeNumber>123</EmployeeNumber>
    <LastName>Chang</LastName>
    <FirstName>Ling</FirstName>
    <JobTitle>Manager</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>60</Rate>
  </Employee>
  <Employee>
    <EmployeeNumber>118</EmployeeNumber>
    <LastName>Jamison</LastName>
    <FirstName>Dick</FirstName>
    <JobTitle>Clerk</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>22</Rate>
  </Employee>
  <Employee>
    <EmployeeNumber>144</EmployeeNumber>
    <LastName>Nakayama</LastName>
    <FirstName>Takeshi</FirstName>
    <JobTitle>Supervisor</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>35</Rate>
  </Employee>
</Employees>
```

### 3.17.1 Edit Input File in Real Time

With your Repository server running, you may edit the input, TimeCard, file in real time and change the data names or values in that file.

- 1 Edit the input file using a text edit application such as WordPad. As a test you can change one or two names in the file.
- 2 Rename the input file whose name has changed to **TimeCard\_Multiple.~in**. Change the file extension name **~in** back to **.xml**.

Within a few seconds the File eWay processes the input file again, **TimeCard\_Multiple.xml**, and writes the results to **s1output1.dat** (**s1output%d.dat** where "d" is incremented with each write—depending on your Multiple Records per File setting. See Figure 41).

- 3 Open the output file with a text edit program and view the data and see your changes.

# Building a Project - Scenario2

This tutorial provides step-by-step procedures for creating and testing an eGate Project. This Project2 addresses a business challenge in which a company uses two systems to handle time tracking and payroll distribution. As the systems use different formats, one system must use data from the other system to accomplish its tasks. This chapter explains how to structure your Project to allow the two systems to work together effectively.

## 4.0.1 Required Resources

See [Chapter 3, ICAN Files Required for the Tutorials](#) on page 16 and [Chapter 3, Sample Data](#) on page 16 of this tutorial.

### This chapter includes

- [“Business Challenge” on page 56](#)
- [“Project Overview” on page 57](#)
- [“Sample Data” on page 60](#)
- [“Create New Project” on page 61](#)
- [“Create Connectivity Map” on page 63](#)
- [“Add Components to the Connectivity Map” on page 65](#)
- [“Create New Object Type Definition” on page 67](#)
- [“Configure Services for Input” on page 72](#)
- [“Apply Business Rules for jcdTimecardToTopic\\_collab” on page 76](#)
- [“Configure Services for Output” on page 77](#)
- [“Apply Business Rules for jcdTopicToPayrollOut\\_collab” on page 81](#)
- [“Apply Collaborations and Link Components” on page 90](#)
- [“Configure eWays and JMS Connections” on page 92](#)
- [“Create Environment” on page 96](#)
- [“Run Bootstrap and Management Agent” on page 100](#)
- [“Run Project2” on page 100](#)
- [“Verify Output Data” on page 101](#)

## 4.1 Business Challenge

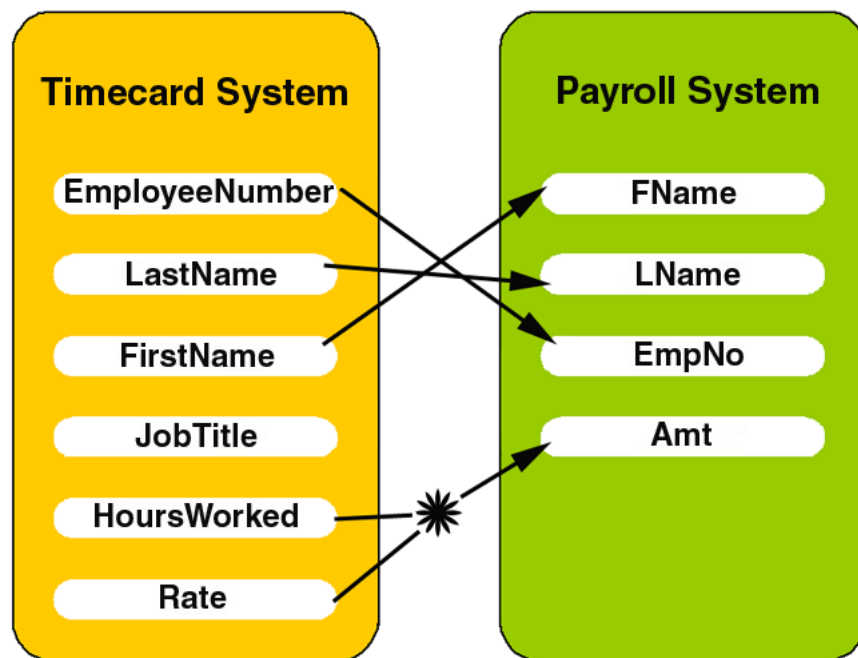
The Project described in this chapter provides a solution to the following business challenge:

- A timecard system tracks the weekly hours worked by employees. The data in this system is in XML format with six fields: **EmployeeNumber**, **LastName**, **FirstName**, **JobTitle**, **HoursWorked**, and **Rate**.
- A Payroll system is responsible for employee payment. The data in this system is in XML format with four fields: **FName**, **LName**, **EmpNo**, and **Amt**. The last element, **Amt**, is calculated by multiplying the **HoursWorked** by the **Rate**.

As shown in Figure 56 below, the following issues must be addressed to successfully integrate the systems:

- The value for the **Amt** field in the Payroll system must be calculated from data in the Timecard system.
- The timecard data must be formatted in such a way that it is recognized by the Payroll system.

**Figure 56** The Relationship Between the Two Systems





## 4.2 Project Overview

Create a basic eGate Project to meet the business challenge described in “**Business Challenge**” on page 56. If you did the Scenario1 tutorial, you may use the same Repository for Scenario2.

### Naming conventions used in this scenario

The following naming conventions are used in scenario2 and are used in SeeBeyond’s training classes.

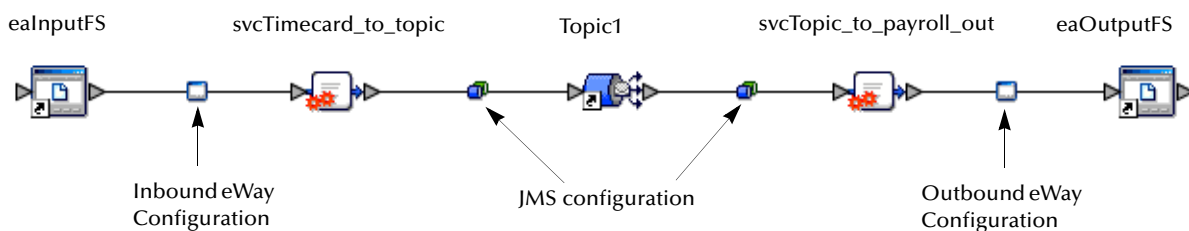
**Table 2** Naming Conventions

Component Type	Naming Convention Prefix
REPOSITORY	<b>REPOSITORY</b>
Project	<b>prj</b> (e.g. <b>prjProject2</b> )
Connectivity Map	<b>cm</b> (e.g. <b>cmCMap2</b> )
Java Collaboration Definition	<b>jcd</b> (e.g. <b>jcdTimecardToTopic_collab</b> )
External Application	<b>ea</b> (e.g. <b>eaInputFS</b> )
Service	<b>svc</b> (e.g. <b>svcTimecard_to_topic</b> )

### 4.2.1 Project Description

The finished eGate Project contains components used to process data in Project2 and publish the data in the appropriate format for the Payroll system.

**Figure 57** Project Connectivity Map

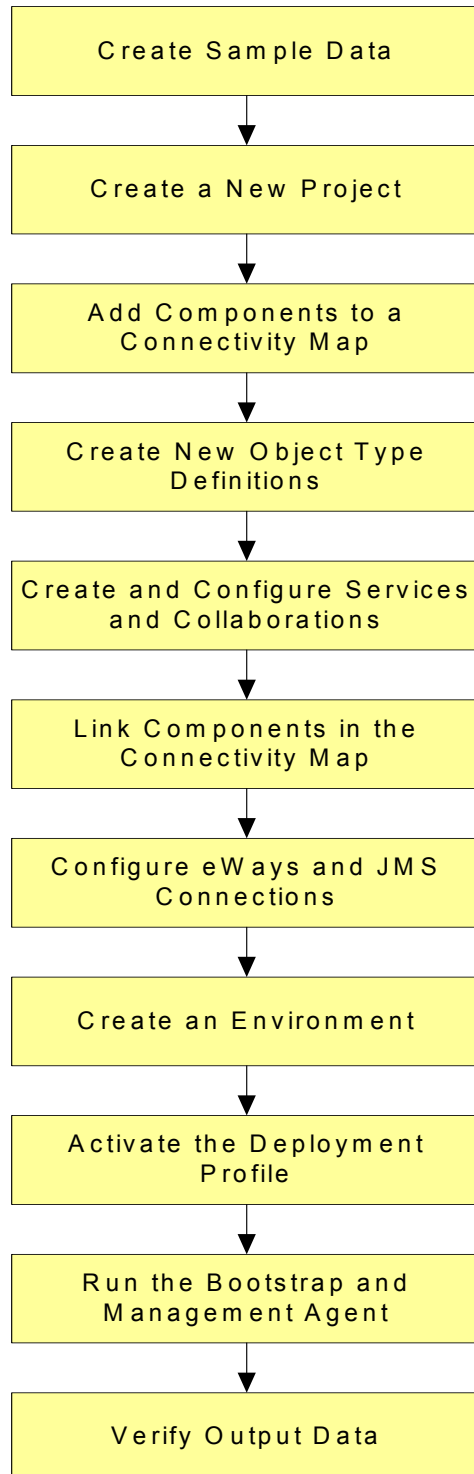


These components perform the following functions:

- **eaInputFS:** The eaInputFS *External Application* represents the Timecard system containing the Input XML data file. (eaInputFS stands for Input File System.)
- **Inbound eWay:** The inbound eWay polls a specified location on the file system for the input XML data file.
- **svcTimecard\_to\_topic:** This Service copies the input data to the **Topic1** without making any changes to the data.

- **Topic1:** This topic publishes the message from the Service **svcTimecard\_to\_topic** to **svcTopic\_to\_Payroll\_out**.
- **svcTopic\_to\_Payroll\_out:** This Service copies the input timecard elements to the appropriate output Payroll system elements. It also calculates the payroll amount.
- **Outbound eWay:** This eWay publishes the finished output data to the file system in the format appropriate for the Payroll system.
- **eaOutputFS:** This represents the Payroll system—the output data file.

**Figure 58** Project Diagram



---

## 4.3 Sample Data

This Project uses the sample files found in the Project2 folder of the **eGate Tutorial Sample**. The files used in Scenario2 are: **s2input.dtd**, **s2output.dtd**, and **Timecard.xml**. To keep it simple, the input file contains only one record. Only one record appears in the output file after you successfully run your project. See [Chapter 3, Sample Data](#) on page 16.

You can also create the files you need. See sections 4.3.1, 4.3.2, and 4.3.3 below.

### 4.3.1 Create Sample Input XML File

Use a text editor such as Microsoft Notepad to create the input XML file. You can also copy and paste from this document (PDF) using the text select tool from within Adobe Acrobat.

#### Create XML file

Your XML file should look similar to the structure and syntax below. In this scenario of the tutorial there is only one record.

- 1 In Windows Explorer, create a directory for the data files. For this sample scenario, use **c:\eGateData\Project2**.
- 2 In Notepad, create an XML file containing the following information:

```
<?xml version="1.0" encoding="UTF-8"?>
  <Employee>
    <EmployeeNumber>100</EmployeeNumber>
    <LastName>Smith</LastName>
    <FirstName>Jane</FirstName>
    <JobTitle>Manager</JobTitle>
    <HoursWorked>40</HoursWorked>
    <Rate>50</Rate>
  </Employee>
```

- 3 Save the file as **Timecard.xml** in the **c:\eGateData\Project2** directory.

### 4.3.2 Create Input (DTD) File

The **s2input.dtd** file is used as a representation of the data in the Timecard system.

The **s2input.dtd** file uses the following elements:

- **EmployeeNumber**
- **LastName**
- **FirstName**
- **JobTitle**
- **HoursWorked**
- **Rate**

## Create input (DTD) File

Your input (DTD) should look similar to the structure and syntax below.

- 1 In Notepad, create an XML file containing the following information:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Employee (EmployeeNumber, LastName, FirstName, JobTitle,
HoursWorked, Rate)>
<!ELEMENT EmployeeNumber (#PCDATA)>
<!ELEMENT LastName (#PCDATA)>
<!ELEMENT FirstName (#PCDATA)>
<!ELEMENT JobTitle (#PCDATA)>
<!ELEMENT HoursWorked (#PCDATA)>
<!ELEMENT Rate (#PCDATA)>
```

- 2 Save the file as **s2input.dtd** in the **c:\eGateData\Project2** directory.

### 4.3.3 Create Output (DTD) File

The **s2output.dtd** file is used as a representation of the data in the Payroll system.

The **s2output.dtd** file uses the following elements:

- **FName**
- **LName**
- **EmpNo**
- **Amt**

## Create Output (DTD) file

Your output (DTD) should look similar to the structure and syntax below.

- 1 In Notepad, create an XML file containing the following information:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Employee (FName, LName, EmpNo, Amt)>
<!ELEMENT FName (#PCDATA)>
<!ELEMENT LName (#PCDATA)>
<!ELEMENT EmpNo (#PCDATA)>
<!ELEMENT Amt (#PCDATA)>
```

- 2 Save the file as **s2output.dtd** in the **c:\eGateData\Project2** directory.

---

## 4.4 Create New Project

There are many ways to create a Project and link the modules. This tutorial presents one scenario. Begin by creating and naming a Project in the Enterprise Designer. The sample Timecard input file used in this scenario contains only one record. The output file contains only one record.

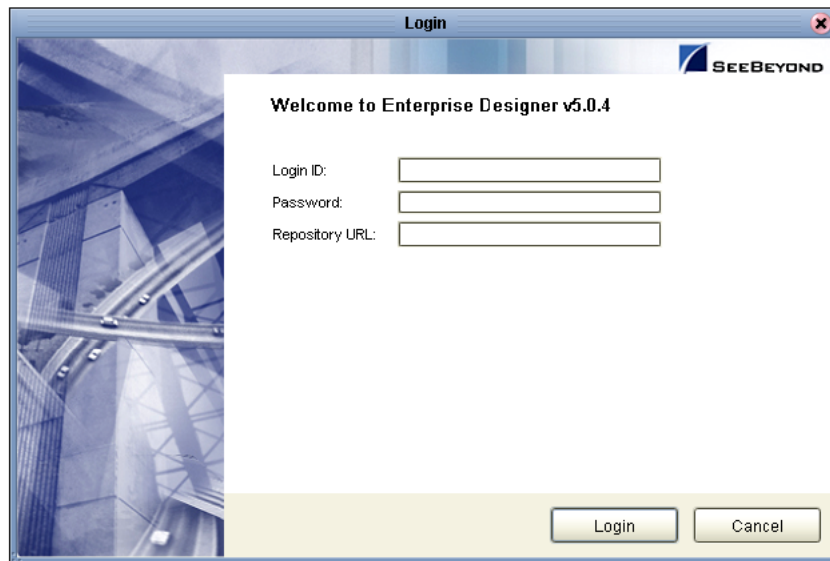
## Start Enterprise Designer

Start the Enterprise Designer according to your organization's instructions.

- 1 Start the Enterprise Designer by executing **runed.bat** in your ican50 folder (c:\ican50\edesigner\bin).

The Enterprise Designer Login dialog box appears.

**Figure 59** Enterprise Designer Login



- 2 Type your **Username** and **Password**. Click **Login** to start the Enterprise Designer.

***Note:** If your login window doesn't appear or if you get an invalid Username or Password error, make sure your repository is running. Also, be aware that the Username and Password are case sensitive.*

## Create prjProject2

A Project is a collection of logical components, configurations, and files that are used to solve a business problem. If you have already done Scenario1, this project could be Project2; otherwise, you could name this project Project1.


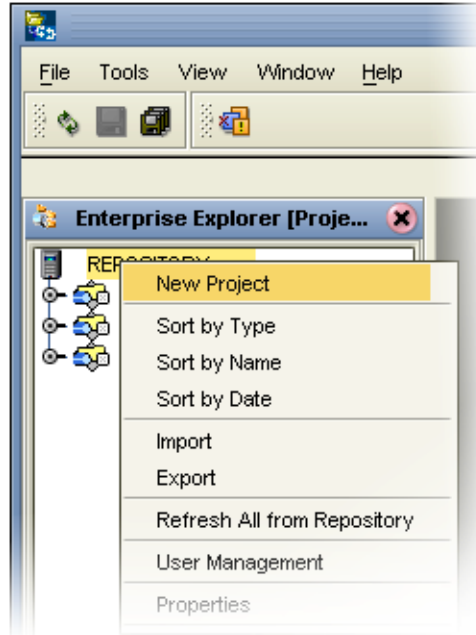
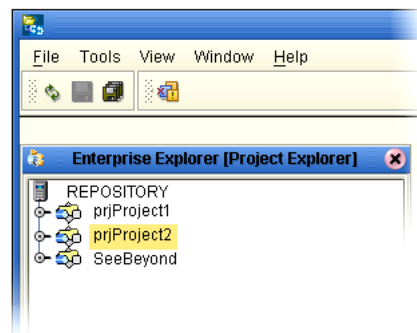
- 1 In the Enterprise Explorer pane of the Enterprise Designer, right-click the Repository name (computer icon ). Click **New Project**.

Figure 60 Create a Project



- 2 Name your Project, **prjProject2** as the file name and press **Enter**.
- 3 The **prjProject2** structure appears in the Explorer pane on the left side of the window.

Figure 61 Project Folder: prjProject2



---

## 4.5 Create Connectivity Map

**prjProject2** needs a Connectivity Map to define the flow between the Timecard system and the Payroll system.

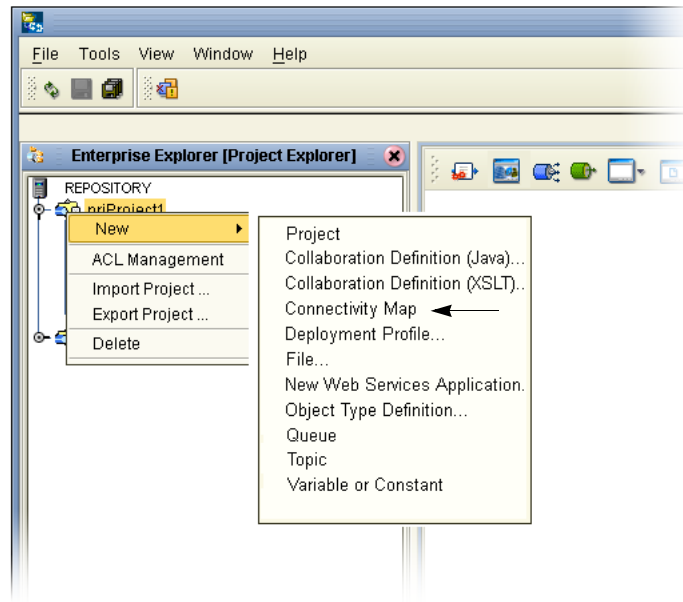
This section describes how to create the Connectivity Map for **prjProject2**.

## Name Connectivity Map

A Connectivity Map contains business logic and routing information about the data transmission. The Connectivity Map is a GUI (graphical representation) of the project.

- 1 In the Enterprise Explorer, right-click the **prjProject2** icon, then click **New**, and **Connectivity Map**.

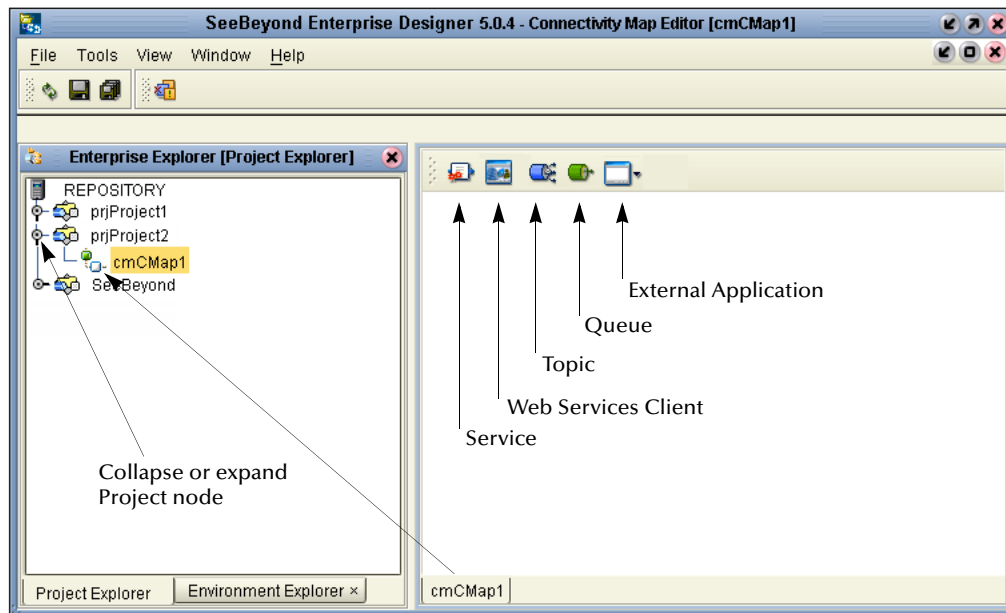
**Figure 62** Create a Connectivity Map



- 2 Name your Connectivity Map, **cmCMap1**.
- 3 **cmCMap1** is added to your **prjProject2**, as shown in the following figure.



Figure 63 prjProject2 with a Connectivity Map (cmCMap2)



*Note:* The **cmCMap1** Connectivity Map appears as an icon in the Project Explorer. A **cmCMap1** tab is also added at the bottom of the Connectivity Map.

---

## 4.6 Add Components to the Connectivity Map

In this section, add the components to the Connectivity Map. Later link the components as shown in [Figure 57 on page 57](#).

### 4.6.1 Add Components to the Connectivity Map

In this section, add the following components to the Connectivity Map:

- **eaInputFS** (File External Application)
- **svcTimecard\_to\_topic**
- **Topic1**
- **svcTopic\_to\_Payroll\_out**
- **eaOutputFS** (File External Application)

#### Create eaInputFS

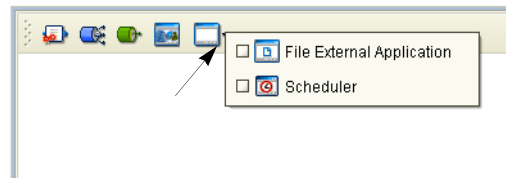
Place an input file (External Application) in your Connectivity Map.

- 1 On the Connectivity Map toolbar, click the External Application arrow (shown in [Figure 64](#)) to display the list. Click **File External Application**.

**Note:** *If the file option doesn't display in the drop-down list, you may need to install FileWay.sar.*

This adds a new **File** icon to the toolbar.

**Figure 64** External Application Selection



- 2 Drag the **File** icon from the toolbar into the left side of the Connectivity Map.  
This creates a new File External Application called **File1**.
- 3 Type or rename the file, **eaInputFS** (external application, input file system). Rename files by Right-clicking on the label.
- 4 Press **Enter**.

**Note:** *Do not type a space between Input and FS. (Input and output are reserved words.)*

#### Create Service, Timecard\_to\_topic

Create a Service to contain binding information about connecting the input Timecard to a JMS topic.

- 1 On the Connectivity Map toolbar, drag the **Service** icon into the Connectivity Map to the right of the **eaInputFS** icon.  
This creates a new Service called **Service1**.
- 2 Right-click the **Service1** label and rename **svcTimecard\_to\_topic**.
- 3 Press **Enter**.

Services are components that interact through messages.

**Note:** *To reposition a component in the Connectivity Map, click the icon and drag to a new location while holding down the mouse button.*

#### Create Topic

The JMS topic conforms to the publish-and-subscribe messaging domain, where one publisher broadcasts messages to potentially many subscribers. (In the tutorials, which are intended to get you started quickly with eGate, topics are not used to their full potential.)

- 1 On the Connectivity Map toolbar, drag the Topic icon into the Connectivity Map to the right of the **svcTimecard\_to\_topic** icon.  
This creates a new topic called **Topic1**. Accept the default name, **Topic1**.

### Create Service, svcTopic\_to\_payroll\_out

Create a Service to contain binding information about connecting the topic to the output payroll file.

- 1 On the Connectivity Map toolbar, drag another **Service** icon into the Connectivity Map to the right of the **Topic1** icon.
- 2 Type or rename, **svcTopic\_to\_payroll\_out**.
- 3 Press **Enter**.

### Create eaOutputFS

Place an output file (external application) in your Connectivity Map.

- 1 Drag the **File** icon into the Connectivity Map, and place it to the right of the **svcTopic\_to\_payroll\_out** icon.  
This creates a new External Application file.
- 2 Right-click the **File** label and rename **eaOutputFS**.
- 3 Press **Enter**.

**Figure 65** Connectivity Map



*Note:* These objects are first configured, and later they will be linked.

---

## 4.7 Create New Object Type Definition

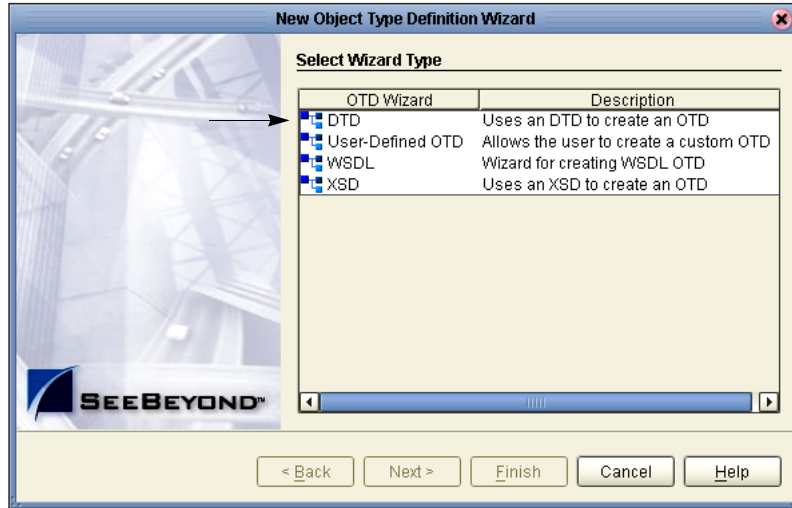
Create an Object Type Definition before you configure your Services. OTDs represent the structure of the data and are used for message parsing; they contain send/receive methods. Use the OTD wizard to make the OTD creation easy.

### Create Object Type Definition

First use DTD to create an OTD from a Document Type Definition.

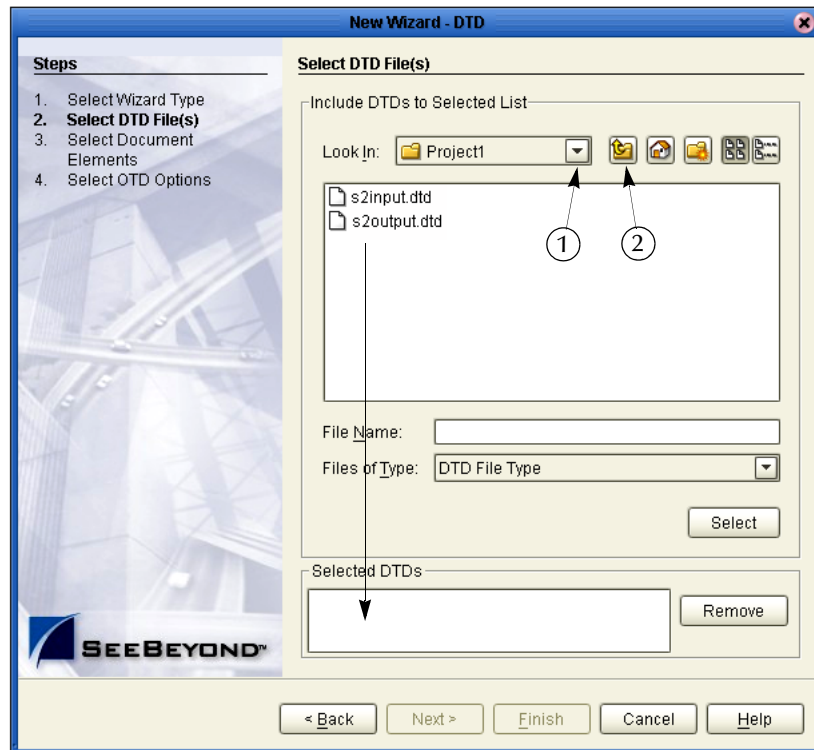
- 1 Right-click **prjProject2**.
- 2 Click **New**, then **Object Type Definition**.
- 3 Click the **DTD** item. (See the following figure.)

Figure 66 OTD Wizard Selection



- 4 Click Next.
- 5 Click the drop-down arrow in the Look In dialog box (number 1 in the following figure).
- 6 Navigate to the sample data folder. For example, `c:\eGateData\Project2`.

Figure 67 Select DTD Files

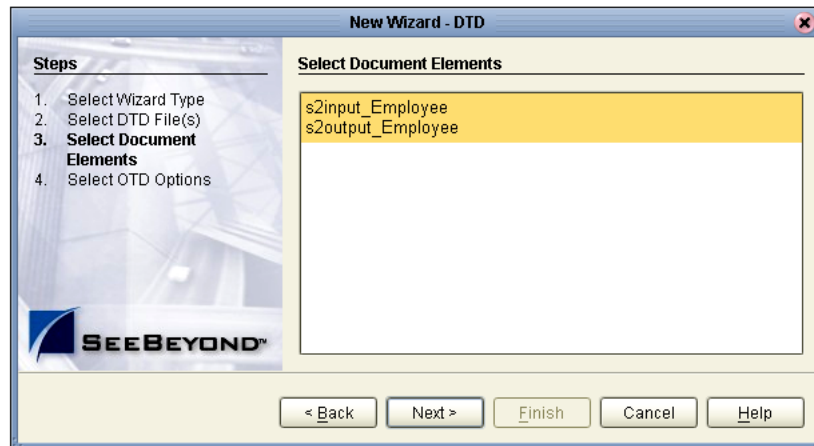


- 7 Highlight `s2input.dtd`. Click **Select** (or double-click).

- 8 Highlight **s2output.dtd**. Click **Select** (or double-click).  
The **s2input.dtd** and **s2output.dtd** should appear in the Selected DTDs.  
(You can also select both at the same time by holding down the shift key.)
- 9 When both **s2input.dtd** and **s2output.dtd** appear in the Selected DTDs, click **Next**.

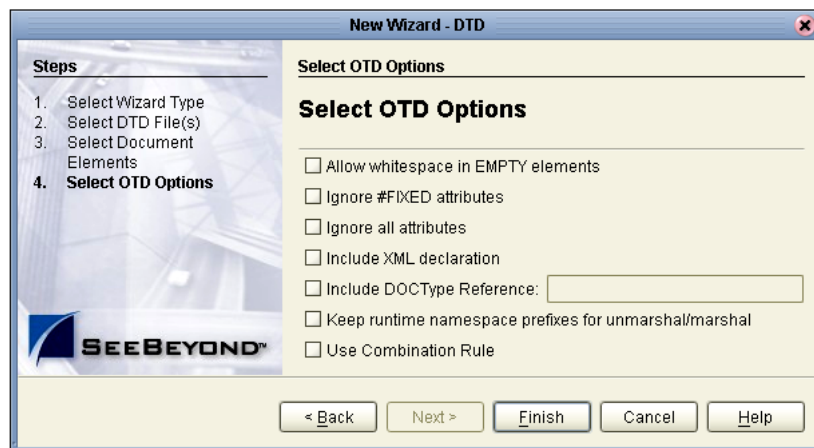
*Note: The **Up one Level** icon (number 2 in Figure 67) is used to navigate to a previous screen; this is useful if you have made an incorrect selection.*

**Figure 68** Select Document Elements



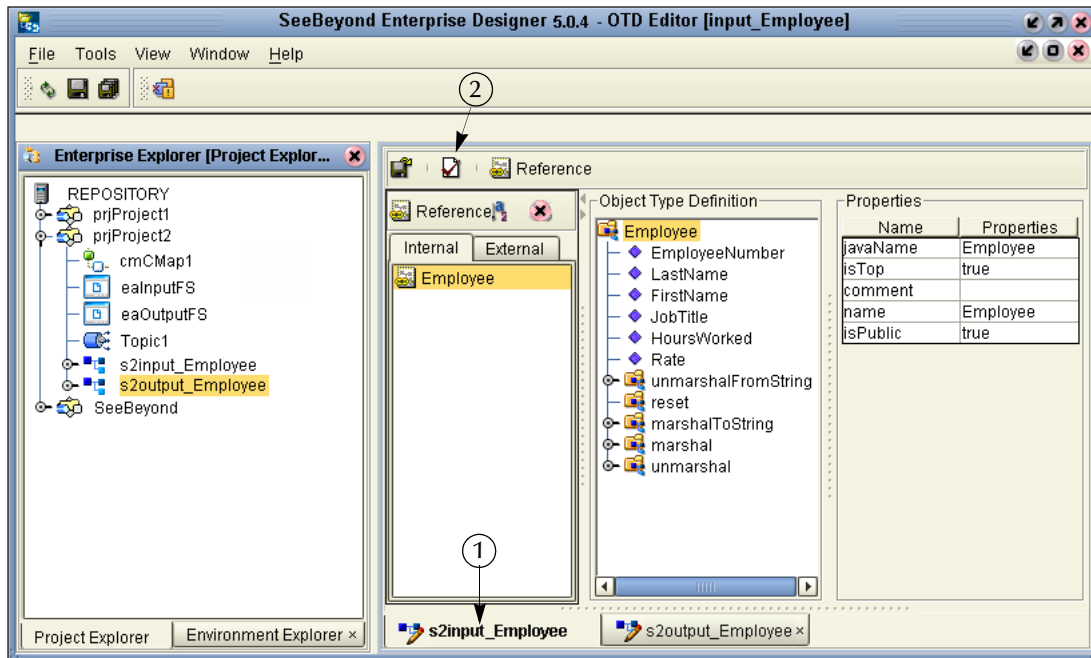
- 10 Select both **s2input\_Employee** and **s2output\_Employee** (hold down the CTRL key while selecting).
- 11 Click **Next**.

**Figure 69** Select OTD Options



- 12 Click **Finish** (accept the defaults).  
The OTD Editor window opens. See the following figure.

Figure 70 OTD Editor Input



13 On the **File** menu, click **Save**.

**Note:** *Save* saves all your changes in your current open editor while **Save All** saves all changes across all editors, including changes in the environments.

### Validate Input Data

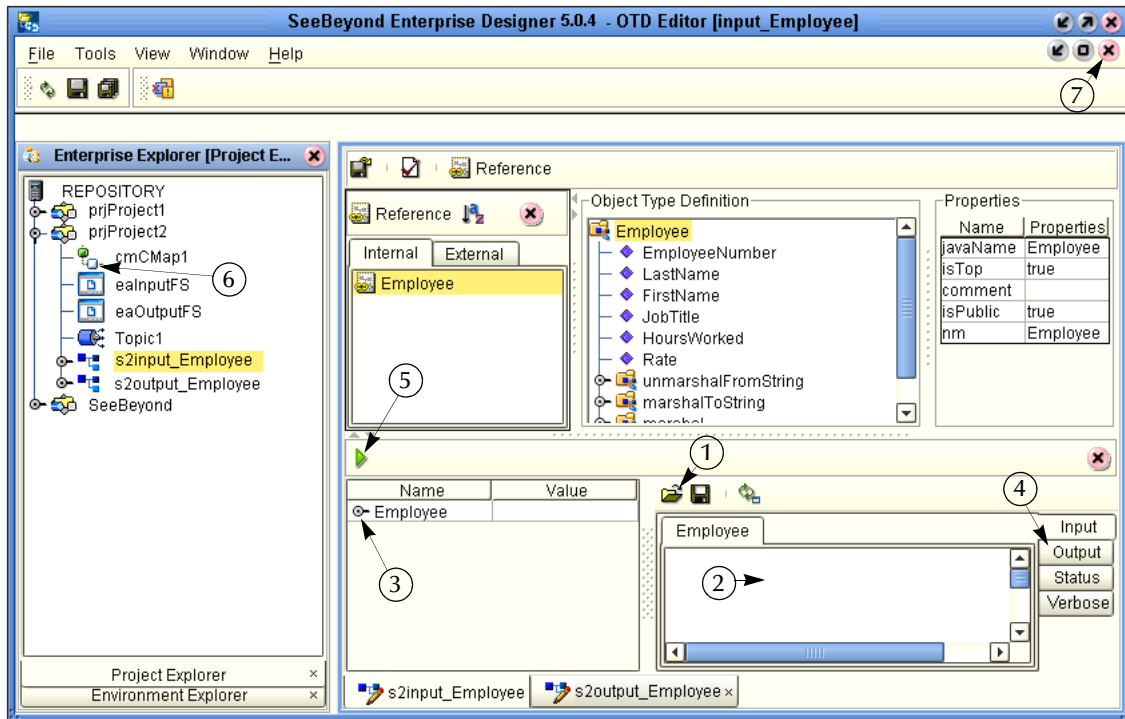
Use the following steps to validate the syntax and run a test to check your DTD files. In the following steps test the input **Timecard.xml** file.

**Important:** Make sure the *s2input\_Employee* tab is selected (number 1 in Figure 70).

- 1 Click the **s2input\_Employee** tab, if it is not already selected.
- 2 Click the **Tester** icon (number 2 in Figure 70 above).

An expanded window appears. See the following figure.

Figure 71 Test Input Data File



- 3 Click the **Open File** icon (number 1 in Figure 71 above) and navigate to your input data (**Timecard.xml**).
- 4 Double-click your sample data file (**Timecard.xml**) to display the XML data in the window. If the file is read successfully the Status “Unmarshal Successful” appears in the Employee window (number 2 in Figure 71).
- 5 Click the expand view icon (number 3 in Figure 71) to expand the view of the root element **Employee**, showing the child elements (fields).
- 6 Click the Output button (number 4 in Figure 71). Output is empty before the test is run.
- 7 Click the green **Tester** icon (number 5 in Figure 71) to execute the test. If the test is successful the Status “Marshal Successful” appears in the Employee window (number 2 in Figure 71).
- 8 Click the Output button again to display the test results. This is not the output file but merely the “output” meaning the results of the test of the input DTD.  
If there are any errors in your file, the Status tab is populated with error messages. The message “Unmarshal Successful” means that the file can be read (unmarshalled). The message “Marshal Successful” means that the file can be written (marshalled).
- 9 Double-click the **cmCMap1** icon (number 6 in Figure 71) to redisplay the Connectivity Map (or close the window, number 7 in Figure 71).

## 4.8 Configure Services for Input

Use Service wizards in the following steps to configure your Collaborations.

### 4.8.1 Configure svcTimecard\_to\_topic

Configure the connectivity of the input Timecard file to the topic.

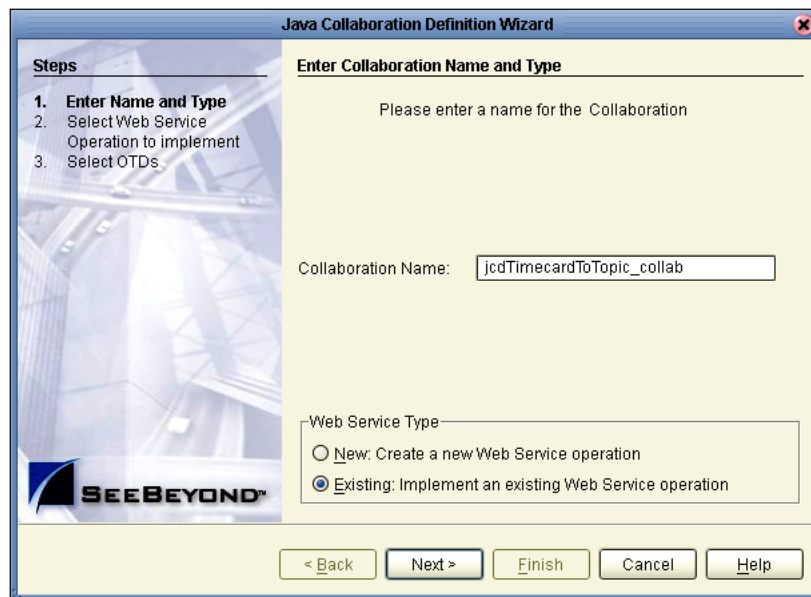
#### Create Web Service Interface

Web Services enable communication between diverse applications using the Internet. Refer to *eGate Integrator User's Guide* for information about Web Services.

- 1 Right-click **prjProject2**.
- 2 Click **New**, then **Collaboration Definition (Java)**.

The Collaboration Definition Wizard (Java) appears.

**Figure 72** New Collaboration Definition (Java) Name



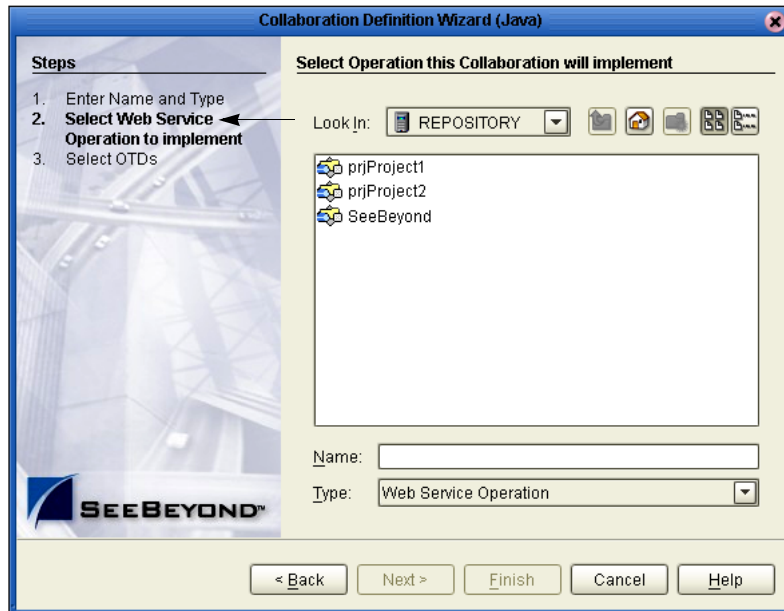
**Note:** Accept the default radio button, **Existing Web Service**, for this Tutorial.

- 3 Type **jcdTimecardToTopic\_collab** as the name for the Collaboration.
- 4 Click **Next**.

**Note:** You must give the collaboration a unique name, not **svcTimecard\_to\_topic**, which is the Service name in the Connectivity Map. **jcdTimecardToTopic\_collab** is the Java Collaboration Definition which appears in the Project Explorer tree. The Service and the Collaboration are bound together later.



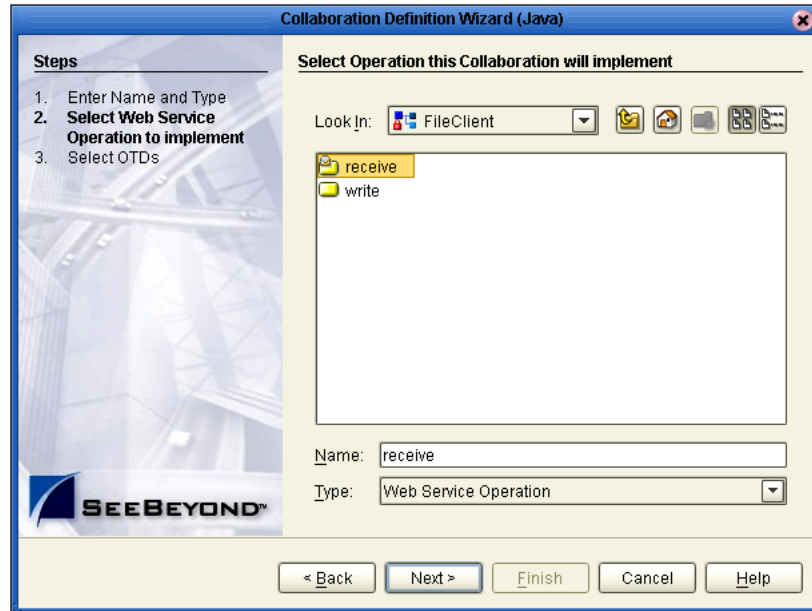
**Figure 73** New Collaboration Definition Wizard (Java)



Select a web service to “implement” a file-read Service, then select an OTD.

- 5 Double-click **SeeBeyond**.
- 6 Double-click **eWays**. (eWays are message end points that connect to an external service provider (system, person, or organization) to enable the sending and receiving of messages.)
- 7 Double-click **File**.
- 8 Double-click **FileClient**.
- 9 Click **receive**. Notice that **receive** appears in the Name field as shown in the following figure.

Figure 74 New Collaboration Web Service Interface



- 10 Click **Next** when “receive” appears in the File Name field.

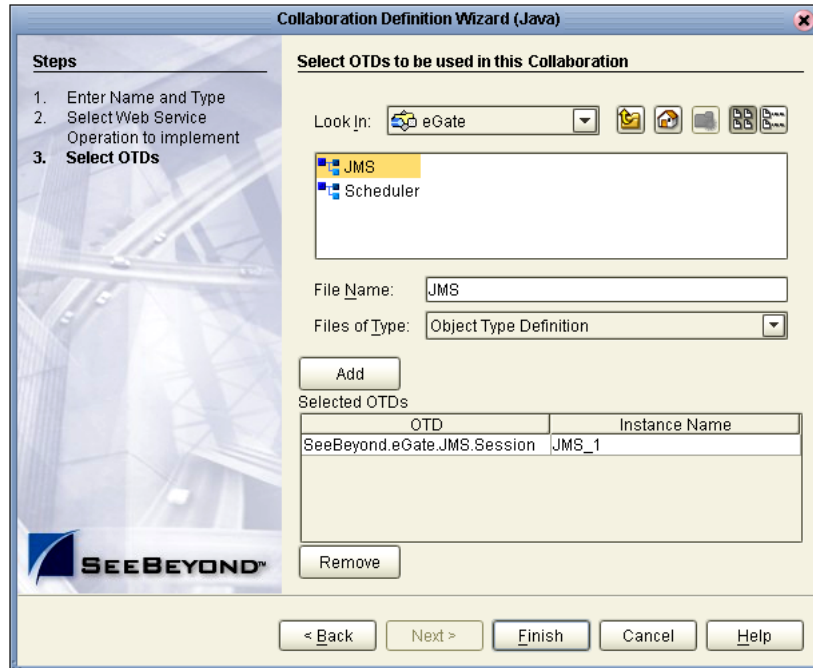
The New Collaboration Definition Wizard (Java) refreshes, and you can select OTDs. The existing web service **FileClient.receive** is used to bring data in.

#### Select OTDs

The following steps set up the OTD for the existing web Service to write to JMS. The **JMS.receive** is used to rout data between Topics (and Queues). You are still using the Collaboration Definition Wizard (Java).

- 1 Double-click **SeeBeyond**.
- 2 Double-click **eGate**.
- 3 Click **JMS**. Click **Add** (or double-click). **JMS** appears in the **Selected OTDs** field as shown in the following figure.

Figure 75 New Collaboration Select OTD

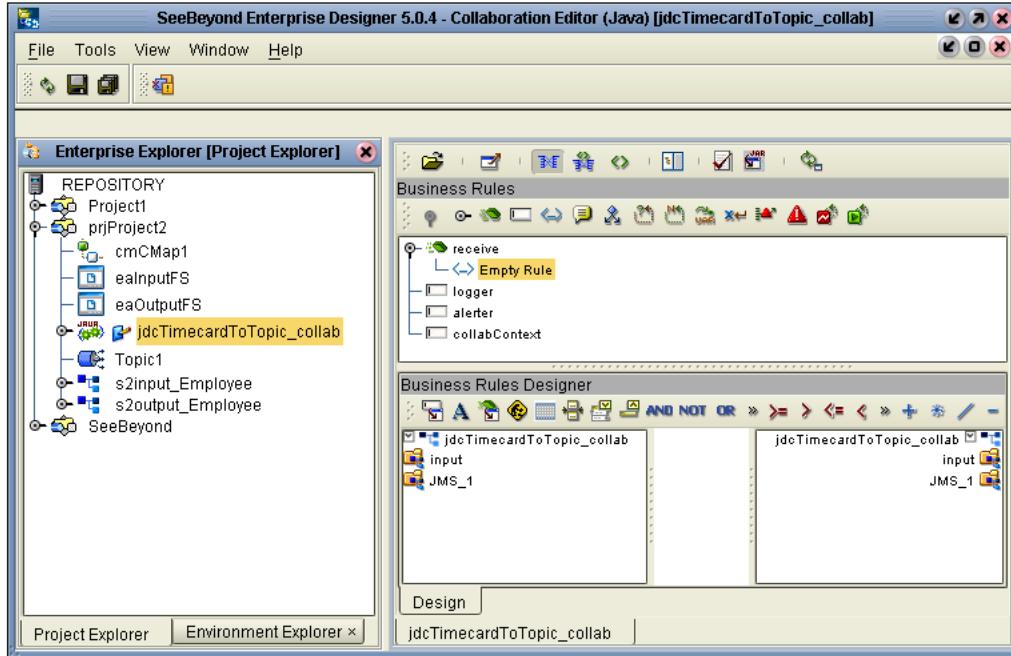


4 Click **Finish**.

After a few seconds, the **jcdTimecardToTopic\_collab** icon appears in the Project Explorer tree in the left pane. Later, this Collaboration is bound to the Service, **svcTimecard\_to\_topic**, in the Connectivity Map, using the drag and drop method.

The Business Rules Designer window appears in the top pane. See the following figure.

Figure 76 Business Rules Designer



## 4.9 Apply Business Rules for jdcTimecardToTopic\_collab

In **jdcTimecardToTopic\_collab** connect the input from the Timecard to the JMS OTD. The JMS passes the message through to the next Service. The processing of the message occurs in the next service, **jdcTopicToPayrollOut\_collab**.

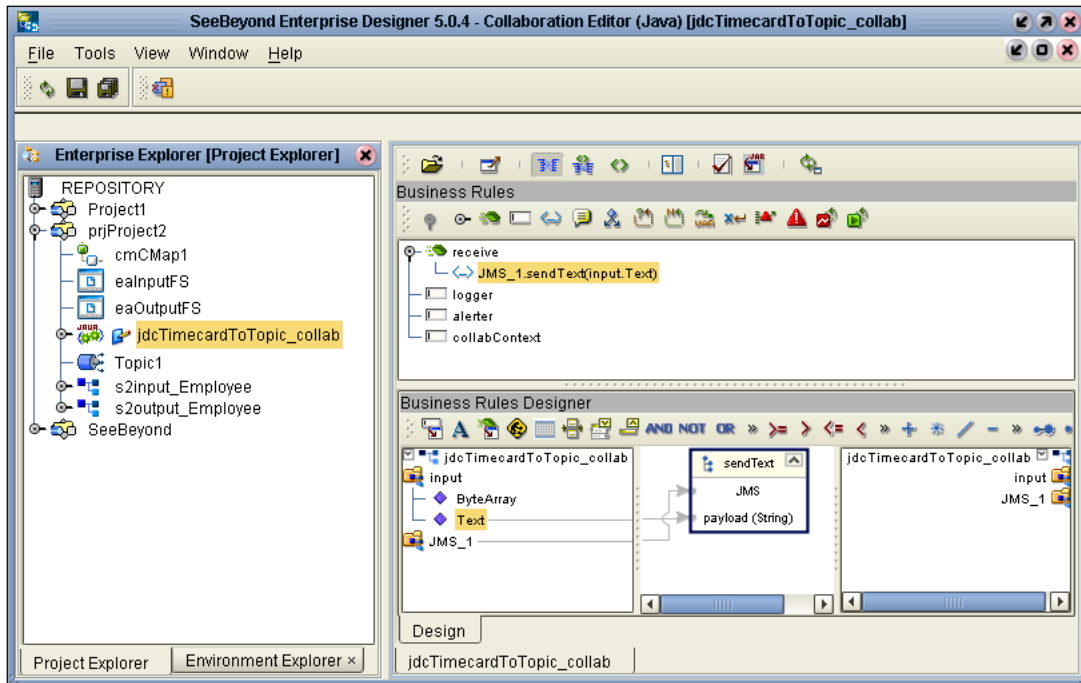
### Send Input to Next Service

Select the Send method so the message can be passed on to the next service.

- 1 Click and drag **JMS\_1** from the left pane to the Business Rules Designer pane.
- 2 Double-click the method to call **sendText(String payload)** from the pop-up dialog box.
- 3 A send **sendText** box appears in the center pane.
- 4 Double-click **input** in the left pane.
- 5 Connect **Text** from **input** to **payload(String)** in the **SendText** box.

See the following figure.

Figure 77 Send Method Box



**Note:** Notice that the objects in Figure 77 reside in both the right and left pane of the Business Rules Designer window. Keep in mind that the GUIs (JMS\_1 for example) refer to the same object.

- 6 On the **File** menu, click **Save All**.
- 7 Click the **Window** menu. Click **Close All**.
- 8 Double-click **cmCMap1** to redisplay the Connectivity Map.

This completes the setup for the **jdcTimecardToTopic\_collab** collaboration definition. You are now ready to drag-and-drop the Service configuration into your Connectivity Map, but first configure **svcTopic\_to\_payroll\_out**.

---

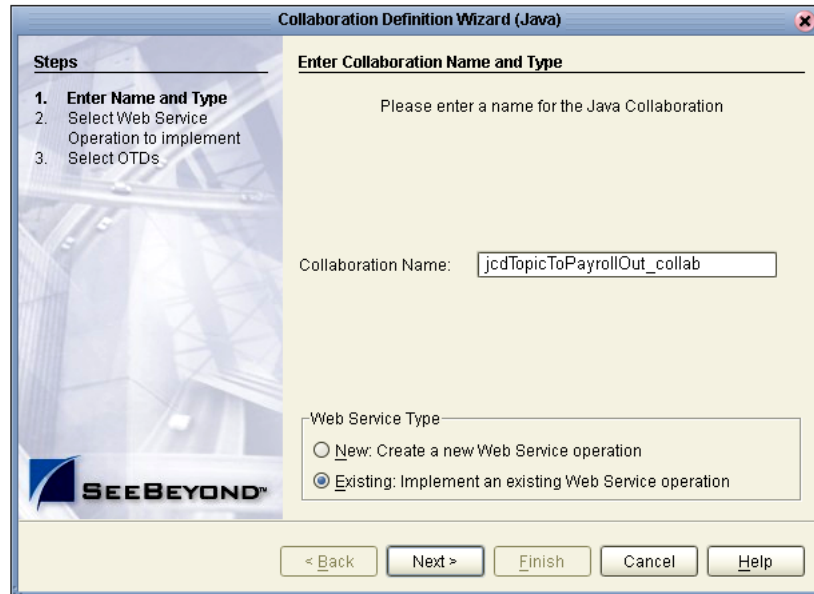
## 4.10 Configure Services for Output

### Select a Web Service Operation

Configure your Web Service interface.

- 1 Right-click **prjProject2**.
- 2 Click **New**, then **Collaboration Definition (Java)**.

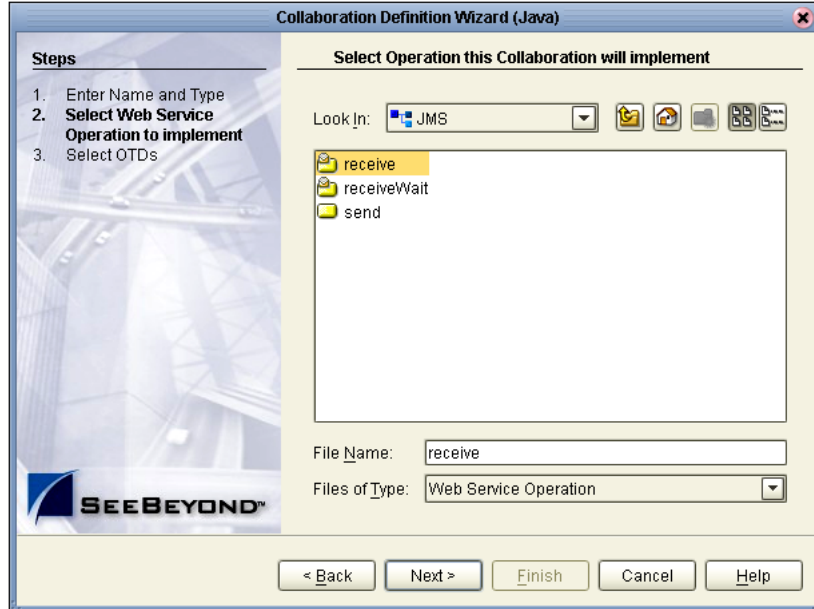
Figure 78 New Collaboration Definition Editor (Java)



*Note:* Accept the default radio button **Existing Web Service** for this Tutorial.

- 3 Type **jcdTopicToPayrollOut\_collab** as the name for the Collaboration Definition (Java).
- 4 Click **Next**.
- 5 From the Collaboration Definition Wizard (Java), double-click **SeeBeyond**.
- 6 Double-click **eGate**.
- 7 Double-click **JMS**. This is your source for this Collaboration.
- 8 Click **receive**. (receive is a method/Web Service used to read a message from a topic that your collaboration implements.)

Figure 79 Receive from Topic



9 Click **Next**.

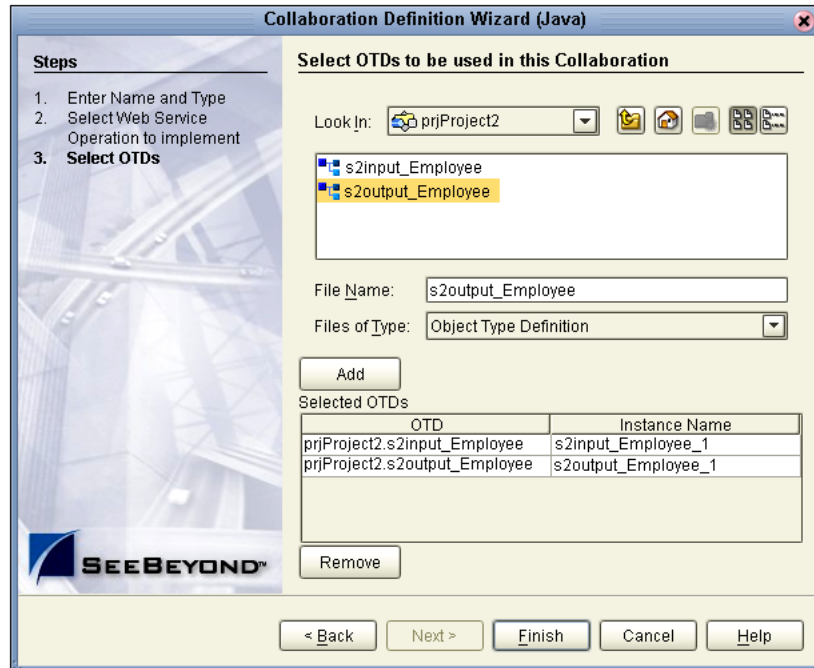
The Collaboration Definition Wizard (Java) refreshes. Now select input and output OTDs.

Select OTDs

1 Double-click **prjProject2** in the Collaboration Definition Wizard (Java).

The Collaboration Definition Wizard (Java) refreshes, showing the input and output options.

**Figure 80** Input/Output OTDs for jcdTopicToPayrollOut\_collab

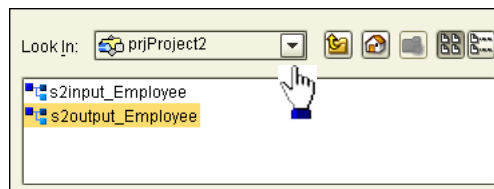


- 2 Double-click **s2input\_Employee**. Double-click **s2output\_Employee** to move these selections to the list of Selected OTDs.

Before clicking Finish, you must first select FileClient which represents your output to an external file.

- 3 Click the **drop-down arrow** icon for the Look In list.

**Figure 81** Drop Down Arrow button

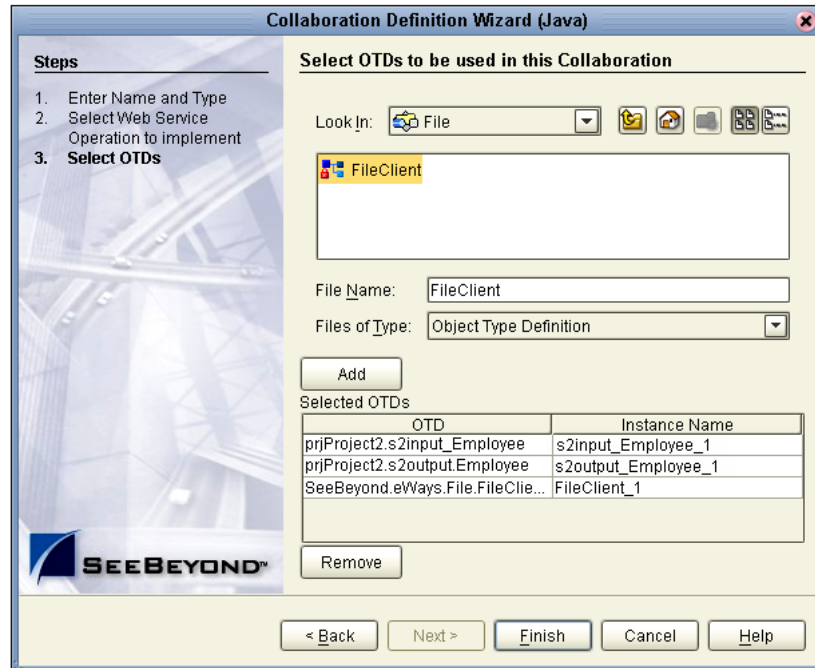


- 4 Click **SeeBeyond**.
- 5 Double-click **eWays**.
- 6 Double-click **File**.
- 7 Double-click **FileClient**.

Notice that the **FileClient\_1** has been added to the list of **Selected OTDs**. See the following figure.



Figure 82 OTDs for jcdTopicToPayrollOut\_collab



8 Click **Finish**.

After a few seconds, the **jcdTopicToPayrollOut\_collab** appears in the Enterprise Explorer tree in the left pane. The Business Rules Designer window also appears, with the Business Rules window above. See the following Figure 83.

---

## 4.11 Apply Business Rules for jcdTopicToPayrollOut\_collab

The following is a summary of what you will do in this section. In this section receive a message from **JMS\_1** and unmarshal the string into OTD **s2input\_Employee\_1**.

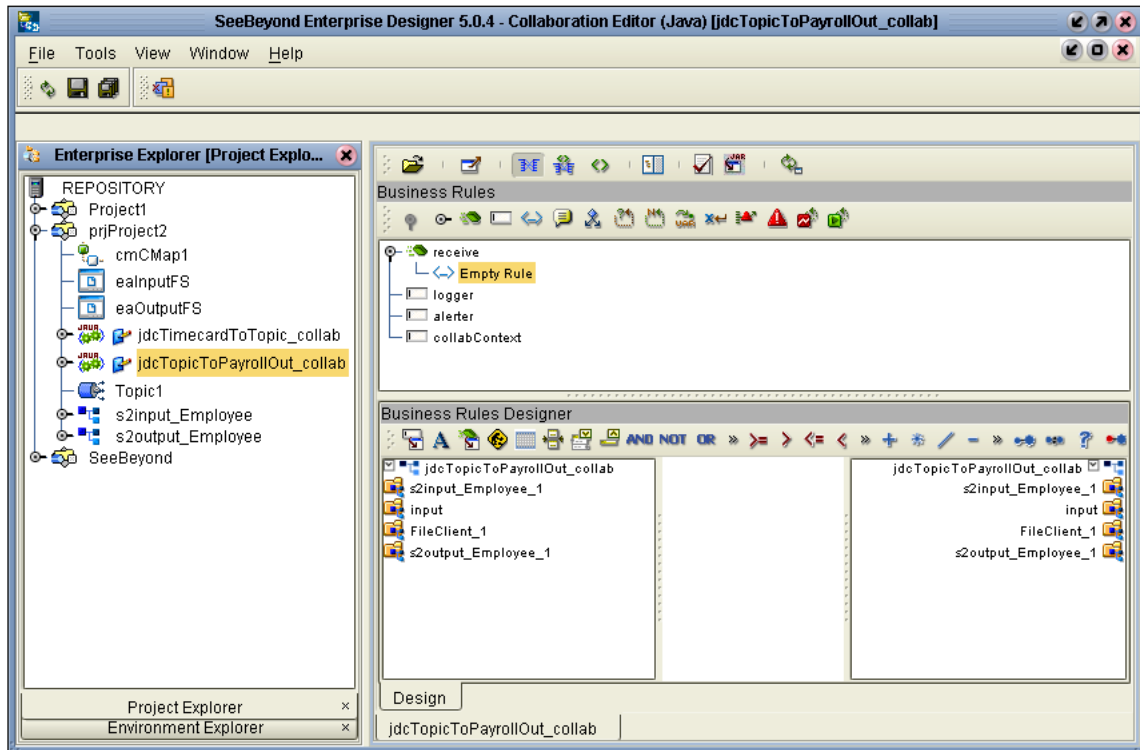
Then copy data from the nodes, **EmployeeNumber**, **LastName** and **FirstName** to the corresponding nodes in **s2output\_Employee\_1** OTD.

Then set up the multiplication logic (business rules) for execution, to obtain an amount (**Amt**) for **s2output\_Employee\_1**, requiring the conversion from string to integer.

Finally, marshal the data from **s2output\_Employee\_1** to a string and write it to a file.

See the following figure.

**Figure 83** Business Rules Designer for jcdTopicToPayrollOut\_collab



### Unmarshal Text

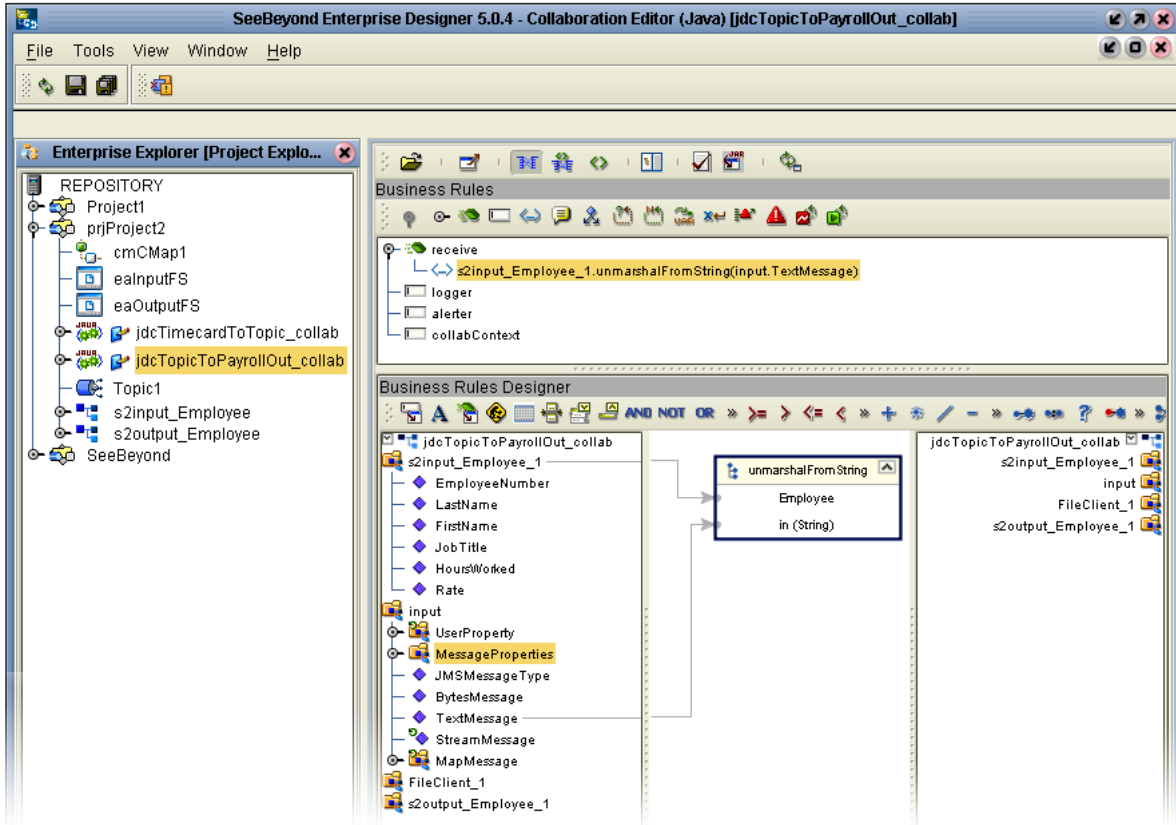
You must first unmarshal the text input from the topic into the input OTD.

- 1 Double-click **input** in the left pane of the Business Rules Designer to expand the view.
- 2 Double-click **s2input\_Employee\_1** in the left pane of the Business Rules Designer to expand the view.
- 3 Click and drag **s2input\_Employee\_1** from the left pane to the Business Rules Designer pane.
- 4 Double-click the method **unmarshalFromString(String in)** from the pop-up dialog box.

This action places a method box into the center pane of the Business Rules Designer window.

- 5 Drag **TextMessage** from **input** into **in (String)**. See the following figure.

Figure 84 Collaboration Editor (Java) - Unmarshal From String



### Connect Input and Output OTD Nodes

Make sure the views of **s2input\_Employee\_1** in the left pane and **s2output\_Employee\_1** in the right pane of the Business Rules Designer are expanded.

- 1 Double-click **s2output\_Employee\_1** in the right pane to expand the view.
- 2 Connect the nodes between the **s2input\_Employee\_1** OTD in the left pane with the **s2output\_Employee\_1** OTD in the right pane of the Business Rules Designer.

Connect the following nodes:

Connect **EmployeeNumber** to **EmpNo**.

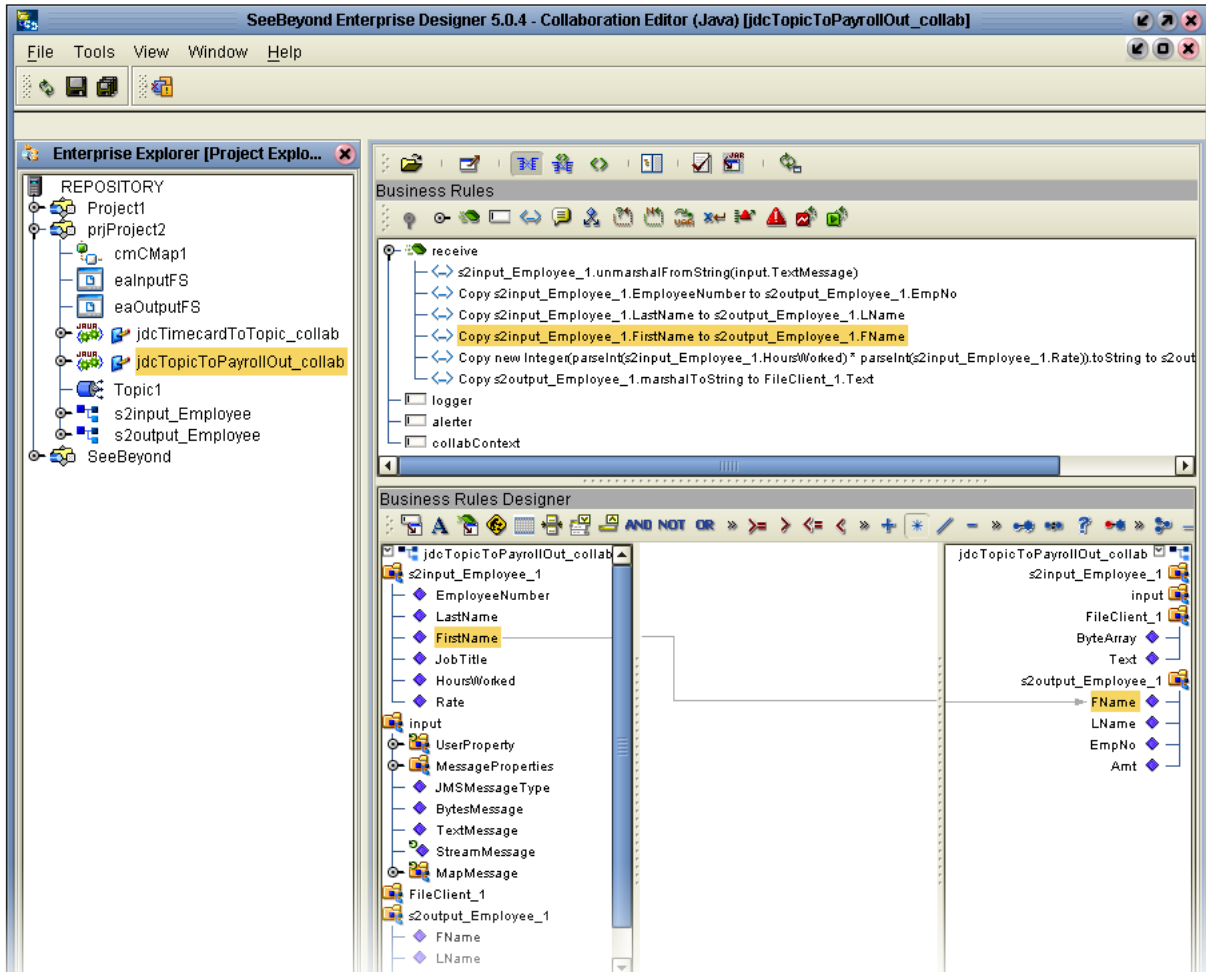
Connect **LastName** to **LName**.

Connect **FirstName** to **FName**.

**Note:** As you connect nodes, the GUI rule disappears, but the record of your action appears in the Business Rules pane which is just above the Business Rules Designer pane.


**Note:** To delete an object, right-click the object and select Delete. You can delete a rule, method box or other GUI object.

Figure 85 Connect Input/Output OTD Nodes



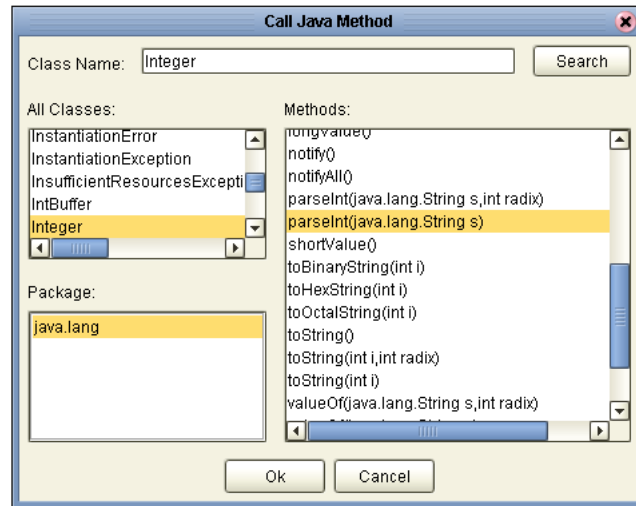
### Create Multiplication Operation

In the following steps, convert string values to Integer so the values can be used in the multiplication operation.

- 1 Click the **Call Java Method** icon  to display a method box.

The method box appears in your center pane. See the following figure.

Figure 86 Integer Java Method



- 2 Type **Integer** (be sure to use capital I) into the Class Name box. Click **Search**.
- 3 Scroll down. Click **parseInt(String s)** from the methods list. Click **OK**.
- 4 Repeat the previous steps because you need two method boxes.

The method boxes may overlap in your workspace. Be sure to locate these boxes and drag them to a location where they do not overlap. See the positioning of the boxes in Figure 88.

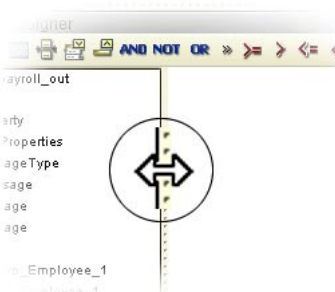
Place these two boxes on the left side of the workspace in the center pane.

- 5 Click the Multiplication Operator **\*** on the toolbar and drag the method box to the right of the two parsing boxes.

The Multiplication Method box appears.

**Note:** You can adjust the size of the window panes by clicking on the frame and adjusting. Refer to the following figure.

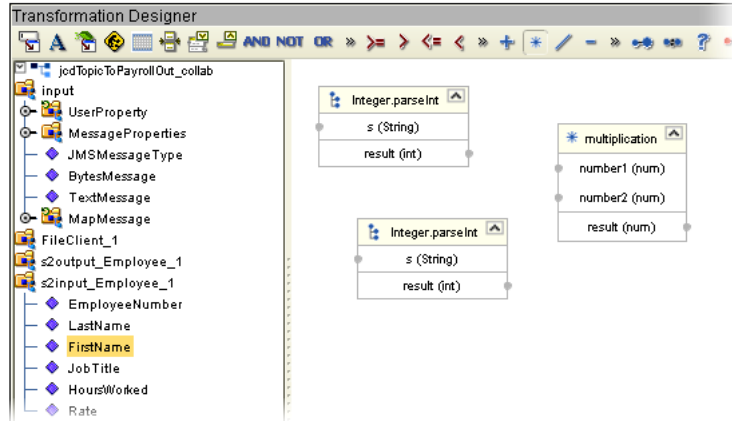
Figure 87 Window Adjust




### Create Method for String to Integer Conversion


Convert int primitive to Integer Object.

**Figure 88** Prepare for Multiplication



- 1 Click the **Call New Constructor**  icon.
- 2 Type **Integer** (be sure to use capital I) and press **Enter**.
- 3 Click **Integer(int value)** in the list. Click **OK**.

A new method box appears. Place this method box under, or to the right of the Multiplication Method box. This method is necessary to convert the int primitive to an Integer Object.

- 4 Click the **Call Java Method** icon  once more to display a method box.
- 5 Type **Integer** (if not already displayed). Click **toString()** in the list.  
Place this method to the right and slightly above the **Integer** dialog box.
- 6 Click **OK**.

This method converts the Integer Object to a String.

#### Connect Nodes

- 1 Connect **HoursWorked** in the **s2input\_Employee\_1** OTD to **s(string)** in the first **Integer.parseInt** method box.
- 2 Connect **Rate** in the **s2input\_Employee\_1** OTD to **s(string)** in the second **Integer.parseInt** method box.

The numbers are now set for conversion to int and ready for multiplication as integers.

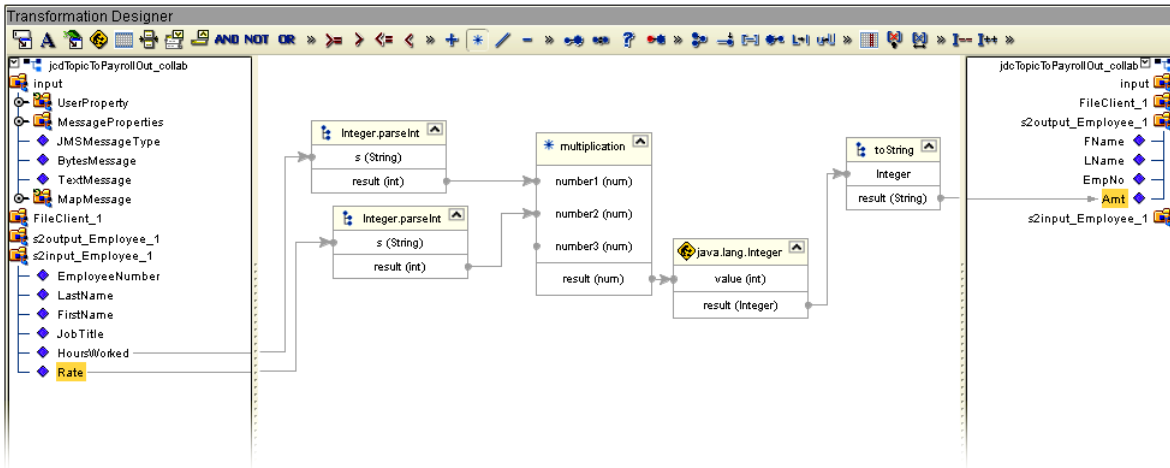
- 3 Connect **HoursWorked** from result to **number1** in the Multiplication method box.
- 4 Connect **Rate** from result to **number2** in the Multiplication method box.
- 5 Connect result from the multiplication to **value(int)** in the **java.lang.Integer**.
- 6 Connect the **result(integer)** to **Integer** in the **toString** method box.
- 7 Connect the **result(String)** to **Amt** in **s2output\_Employee\_1** in the right pane.

Don't select a method at this time.

8 On the **File** menu, click **Save All**.

This completes the multiplication. See the following figure.

**Figure 89** Multiplication Operation



### Marshal Text

Since you began the data manipulation process by unmarshalling data from string, you must now marshal the data from output into a string.

Make sure the view of **FileClient\_1** in the right pane is expanded.

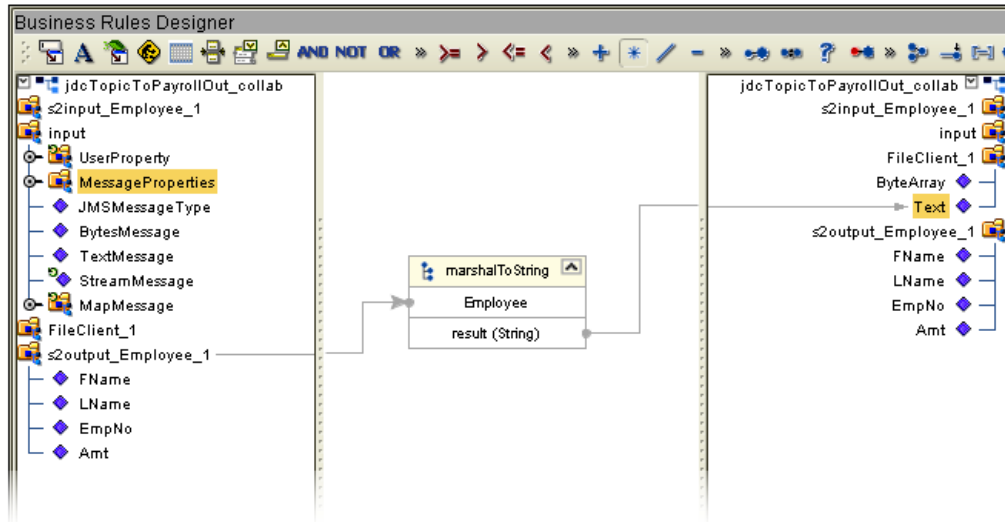
- 1 Click and drag **s2output\_Employee\_1** from the left pane to the Business Rules Designer pane.
- 2 Double-click the method **marshalToString()** from the pop-up dialog box.

The **marshalToString** method box appears.

- 3 Connect a node from **result (String)** to Text under **FileClient\_1** in the right pane. See the following figure.

Don't select a method at this time.

Figure 90 Marshal Text



### Write FileClient\_1

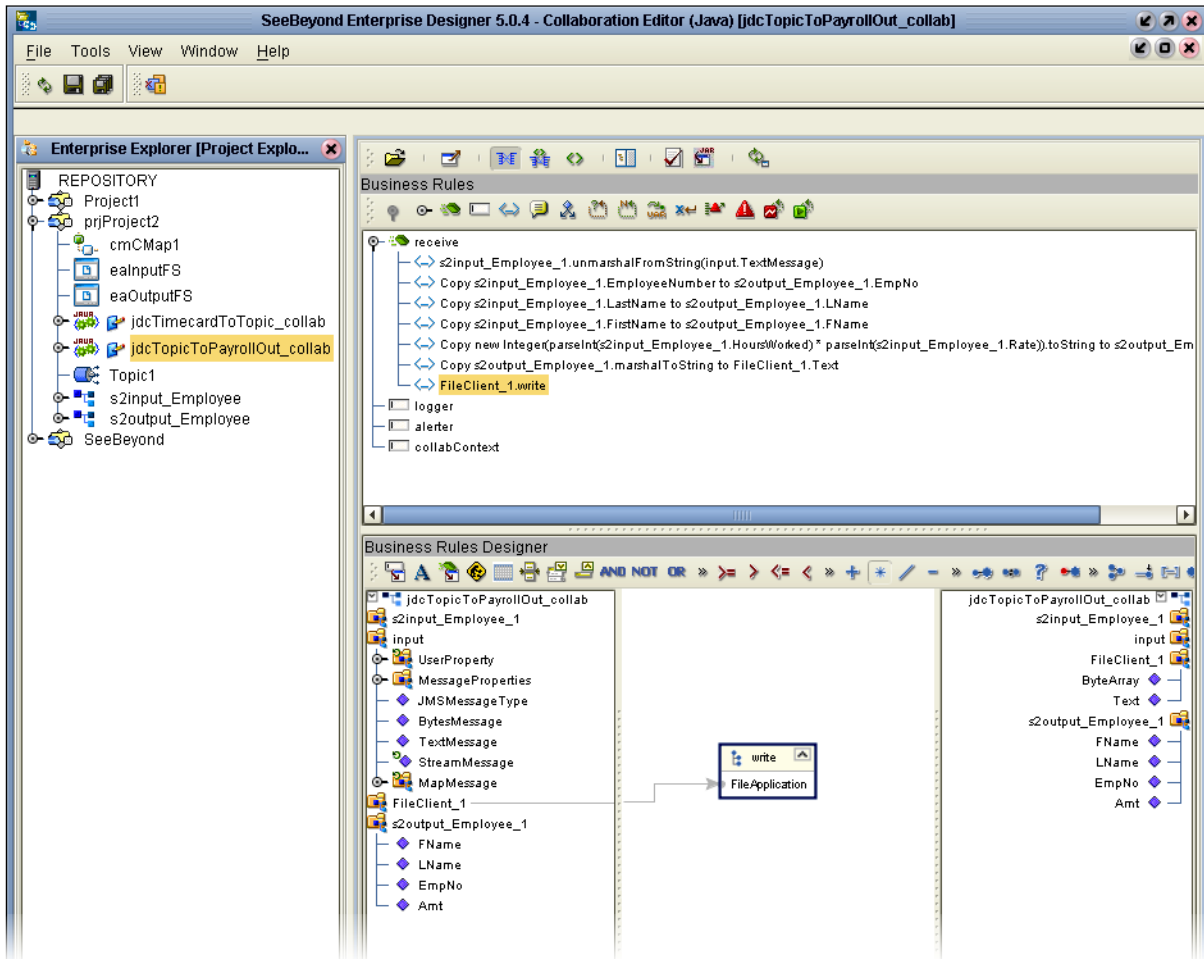
As a final step use the OTD **FileClient\_1** to write your output file.

- 1 Click and drag **FileClient\_1** from the left pane to the Business Rules Designer pane.
- 2 Double click **write ()**.

See the following figure.



Figure 91 Write File Application



A write method box appears with an arrow pointing to FileApplication. This instructs the system to write to the output file.

- 3 On the **File** menu, click **Save All**.

This completes **jcdTopicToPayrollOut\_collab** using the Business Rules Designer window.

Use the drag-and-drop method to move the configurations into your Services on the Connectivity map.

- 4 Navigate back to the Connectivity Map if it is not already displayed. You can close the current window or you can double-click the Connectivity Map name, **cmCMap1**.

## 4.12 Apply Collaborations and Link Components

You have configured your Collaboration Definitions (business rules) in the previous sections. You are now ready to bind your Collaboration Definitions with Services using the drag-and-drop method.

### 4.12.1 Link Components in the Connectivity Map

Linking the components creates the logical flow of data throughout the Project. Additionally, creating a link between an External Application and a Service adds an eWay to the Connectivity Map.

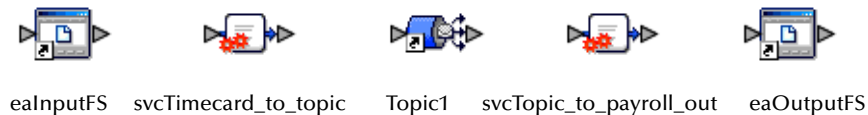
At this point, the Connectivity Map contains five unlinked objects. You are now ready to link them.

#### Drag-and-drop jcdTimecardToTopic\_collab

Refer to Figure 93 for the following steps.

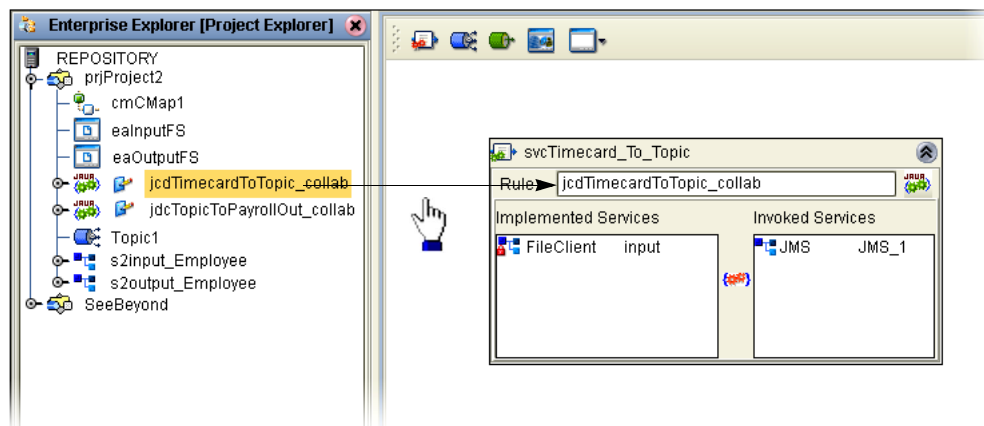
- 1 Double-click the Connectivity Map icon (if the Connectivity Map is not already displayed) to return to the Connectivity Map.

**Figure 92** Unlinked Objects in Connectivity Map



- 2 Double-click the Service **svcTimecard\_to\_topic** in the Connectivity Map to open the Service Window.
- 3 Drag the Collaboration **jcdTimecardToTopic\_collab** from the Project Explorer pane to the **Rule** field in the open window as shown in the following figure.

**Figure 93** Using the Drag and Drop Method

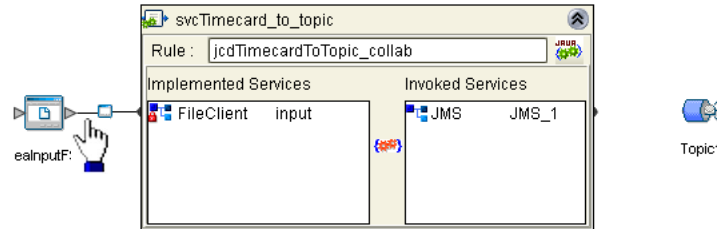


Notice that the gear icons change from red to green to signify that the Collaboration is complete.

**Note:** You may also drag `jcdTimecardToTopic_collab` into the icon without opening the Collaboration window.

- 4 Connect the `eaInputFS` to `FileClient`.

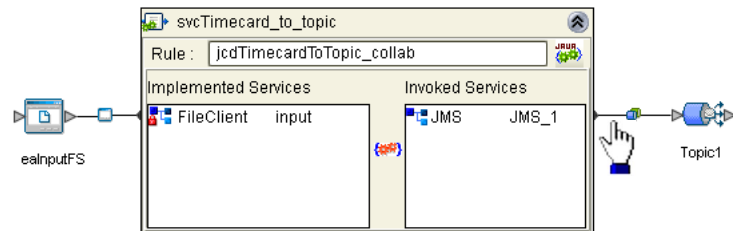
**Figure 94** Connect Input to Client



**Note:** You may connect nodes forward or backward. As long as the connection is made, it makes no difference; however, it may be easier to connect from the Service dialog box backward to the module icon, especially when there are multiple Services.

- 5 Connect `JMS_1` to `Topic1`. (Reposition binding box as necessary.)

**Figure 95** Connect JMS to Topic



- 6 Click the close button (⌵) to close the binding box.

**Figure 96** svcTimecard\_to\_topic Linked

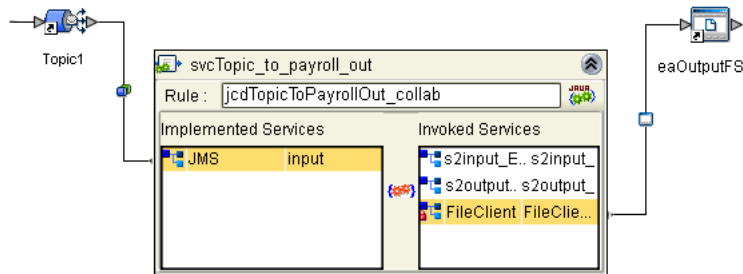


### Drag-and-drop `jcdTopicToPayrollOut_collab`

- 1 Drag the Collaboration `jcdTopicToPayrollOut_collab` from the Project Explorer pane into the Service `svcTopic_to_payroll_out`.
- 2 Double-click the Service `svcTopic_to_payroll_out` to open the binding box.

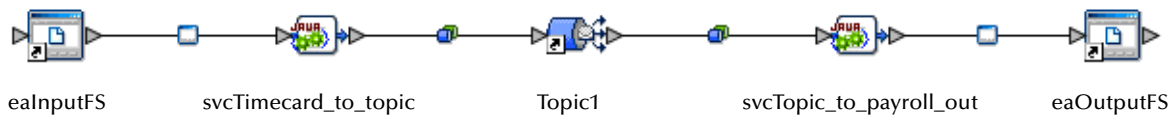
- 3 Connect **Topic1** to **JMS**.
  - 4 Connect **FileClient\_1** to **eaOutputFS**.
- See the following figure.

**Figure 97** Connect Topic to Output



- 5 Click the **close** button (⌵) to close the binding box.

**Figure 98** Linked Connectivity Map



## 4.13 Configure eWays and JMS Connections

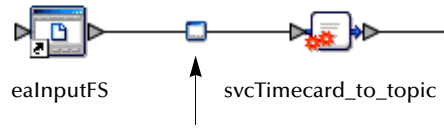
Your Connectivity Map contains two eWays and JMS connections. These components must be configured for deployment. Each of these components performs a different function:

- **Inbound eWay:** The inbound eWay polls a directory on the local file system, **c:\eGateData\Project2**, for any XML file (\*.xml).
- **svcTimecard\_to\_topic:** This Collaboration Service copies the input XML data without making any changes to it.
- **svcTopic\_to\_payroll\_out:** This Collaboration Service performs the calculation and data mapping function.
- **Outbound eWay:** The outbound eWay writes the results to an output file (.dat) on the local file system, **c:\eGateData\Project2**.

### Configure Inbound eWay

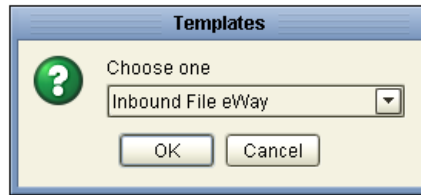
Choose inbound, and identify the inbound file and location.

**Figure 99** Inbound eWay



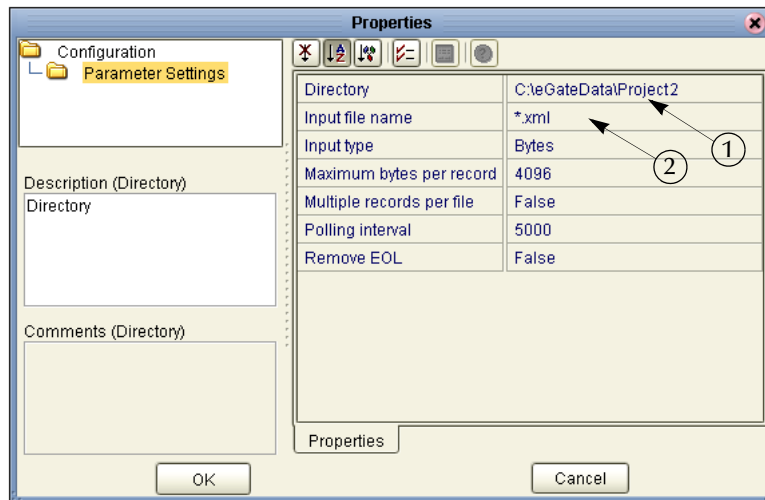
- 1 Double-click the first eWay . Select **Inbound File eWay** from the drop-down list, then click **OK**.

**Figure 100** External Type Dialog Box



This creates a new inbound eWay and the Properties dialog box appears.

**Figure 101** Properties Configuration (Inbound eWay)

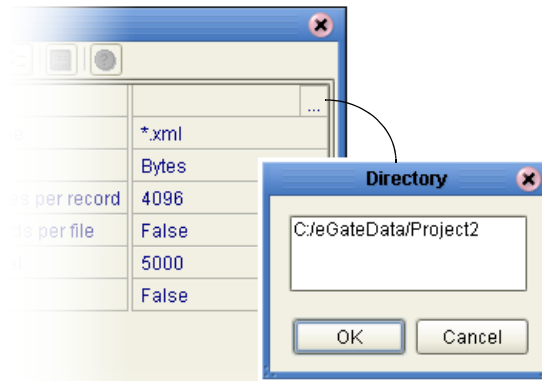


Refer to the arrows in the previous figure. These are the editable fields you configure.

- 2 Click once on the current path to select and type a correct directory path. In this tutorial the path is `c:\eGateData\Project2` (number 1 in Figure 101 above).

Alternately, you can click the ellipses (...) and enter the data path. See the following figure.

Figure 102 Directory Data Path



**Note:** You can use a forward slash (/) or a backward slash (\) for your directory path, but you must be consistent.

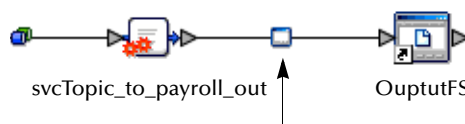
- 3 Click once on the current **Input File Name** field and enter \*.xml. The eWay reads any files with the .xml extension (number 2 in Figure 101).
- 4 Click **OK** to close the Properties Dialog Box for the inbound eWay.

**Note:** If the **Polling interval** value is left unchanged, the eWay polls the directory location every five seconds (5000 milliseconds).

### Configure Outbound eWay

Choose outbound, and identify the outbound file and location.

Figure 103 Outbound eWay



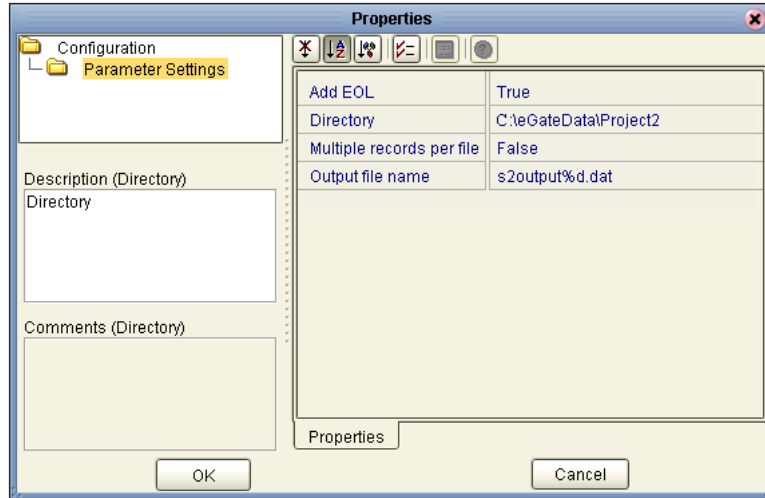
- 1 Double-click the second eWay. Select **Outbound File eWay**. Click **OK**.

The Properties dialog box appears.

Use the same path you used for the inbound eWay (c:\eGateData\Project2). Name the output file name: **s2output%d.dat**.

- 2 Click **OK** to close the Properties Dialog Box for the outbound eWay.

**Figure 104** Properties Configuration (Outbound eWay)



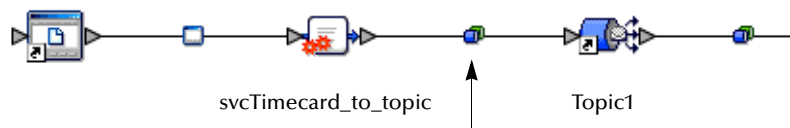
**Note:** The **Multiple records per file** setting determines whether records are concatenated in the output file. A True setting (default) means that multiple records are written in the same file. A False setting, as shown in Figure 104 above, means that only one record is written to one file. See **“Multiple records per file (True or False)”** on page 44 in Scenario1.

You have now configured both the input and output eWays.

## JMS Client Configuration

The JMS Client Configuration icons appear on both sides of the **Topic1** icon.

**Figure 105** JMS Client Properties Icon



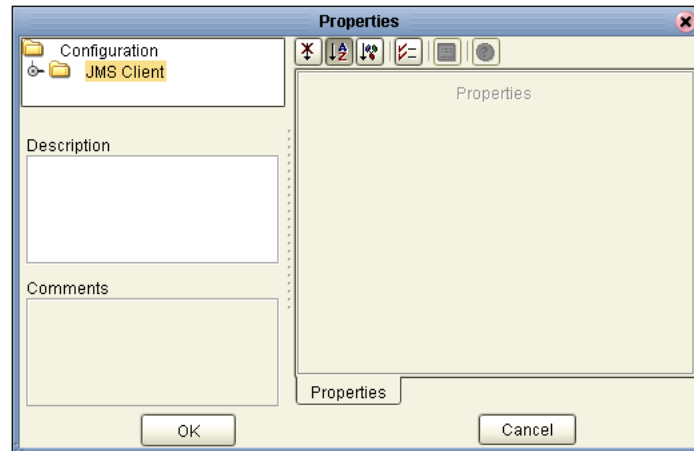
### Select JMS Client Configuration

For this tutorial accept the default configurations.

- 1 Double-click the first of the two JMS icons  (see previous Figure 105).

The Message Server properties dialog box appears.

**Figure 106** Message Server Configuration



- 2 Click **OK** to accept the defaults.
- 3 Repeat this step for the other JMS client configuration.

You are now ready to deploy your Project components in an Environment containing a Logical Host and External Applications.

- 4 On the **File** menu, click **Save All**.  
Your project is now complete and ready for deployment.

---

## 4.14 Create Environment


In this section, create and activate the Deployment Profile. Before you can create the profile, you must create an Environment.

### Create Environment and Logical Host

An Environment is a collection of physical resources and their configurations that are used to host Project components. An Environment contains logical hosts and external systems.

- 1 Click **View** from the Menu bar, then click **Environment Explorer**.

**Note:** *If you have already created an Environment called Tutorial, you may use that Environment and logicalhost. Continue with “[Create External File System for the Inbound eWay](#)” on page 97. Otherwise, continue with the steps in this section.*

- 2 Right-click the Repository name (  ). Click **New Environment**.
- 3 Right-click **Environment1** and rename **Tutorial**. Press **Enter**.
- 4 Right-click **Tutorial**. Click **New Logical Host**.



- 5 Accept the default name **LogicalHost1**.

This creates a **LogicalHost1** window in the right pane.

### Create Integration Server

An integration server is a J2EE software platform that houses the business logic container used to run Collaborations and JCA connectors (eWays).

- 1 Right-click **LogicalHost1** in the **Enterprise Explorer** window. Click **New SeeBeyond Integration Server**.

**IntegrationSvr1** appears in the **LogicalHost1** window.

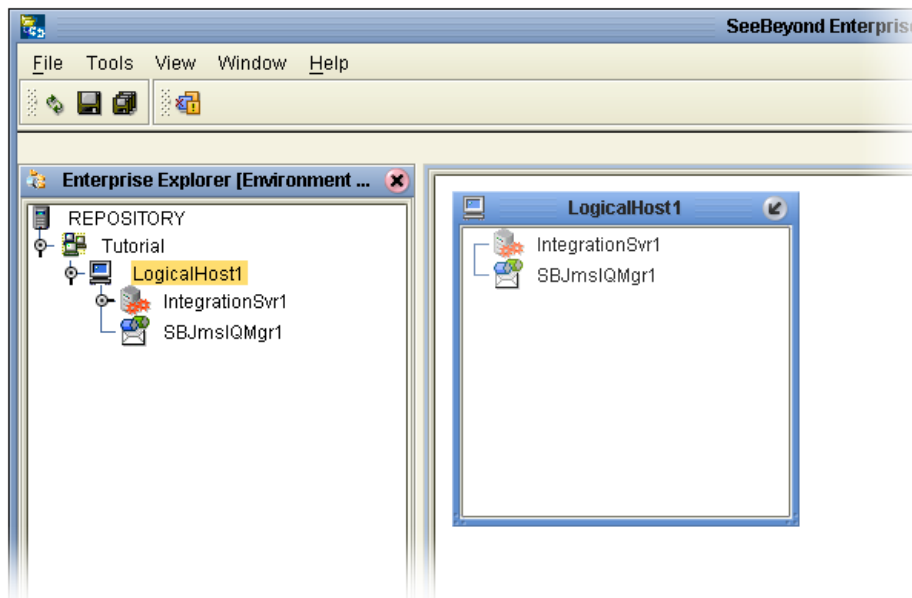
### Create JMS IQ Manager

The JMS IQ Manager is a JMS-compliant, guaranteed delivery store, forwarding, and queueing service.

- 1 Right-click **LogicalHost1** in the **Enterprise Explorer** window. Click **New SeeBeyond JMS IQ Manager**.

**SBJmsIQMgr1** appears in the **LogicalHost1** window.

**Figure 107** Create Logical Host Servers



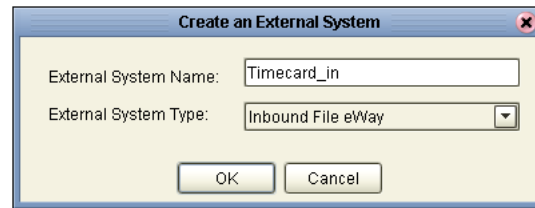
These servers automatically appear in the **LogicalHost1** dialog box.

### Create External File System for the Inbound eWay

If you already have external files from Scenario1 in your Environment pane, you can minimize those dialog boxes.

- 1 Right-click **Tutorial**. Click **New File External System**.

**Figure 108** External File Inbound



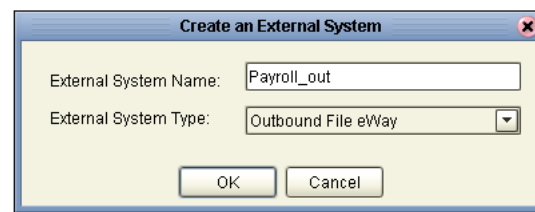
- 2 Type **Timecard\_in** in the name field.
- 3 Select **Inbound File eWay**. Click **OK**.

This places the inbound external system, **Timecard\_in** in the Environment Editor pane.

#### Create External File System for the Outbound eWay

- 1 Right-click **Tutorial**. Click **New File External System**.

**Figure 109** External File Outbound



- 2 Type **Payroll\_out** in the name field.
- 3 Select **Outbound File eWay**. Click **OK**.

This places the outbound external system, **Payroll\_out** in the Environment Editor pane.

**Note:** If the “Create an External System” (or any Java pop-up) window doesn’t appear, it may be hidden behind your Enterprise Explorer window. Press the ALT+TAB keys to see if you have any hidden pop-ups. Use this keystroke combination if your Enterprise Explorer GUI appears to be “frozen.”

### 4.14.1 Create and Activate Deployment Profile

First create a Deployment Profile, then deploy (map) the components.

**Note:** If you already created *Deployment1* for *Scenario1*, you need a different Deployment profile for *Scenario2*, e.g. *Deployment2*.

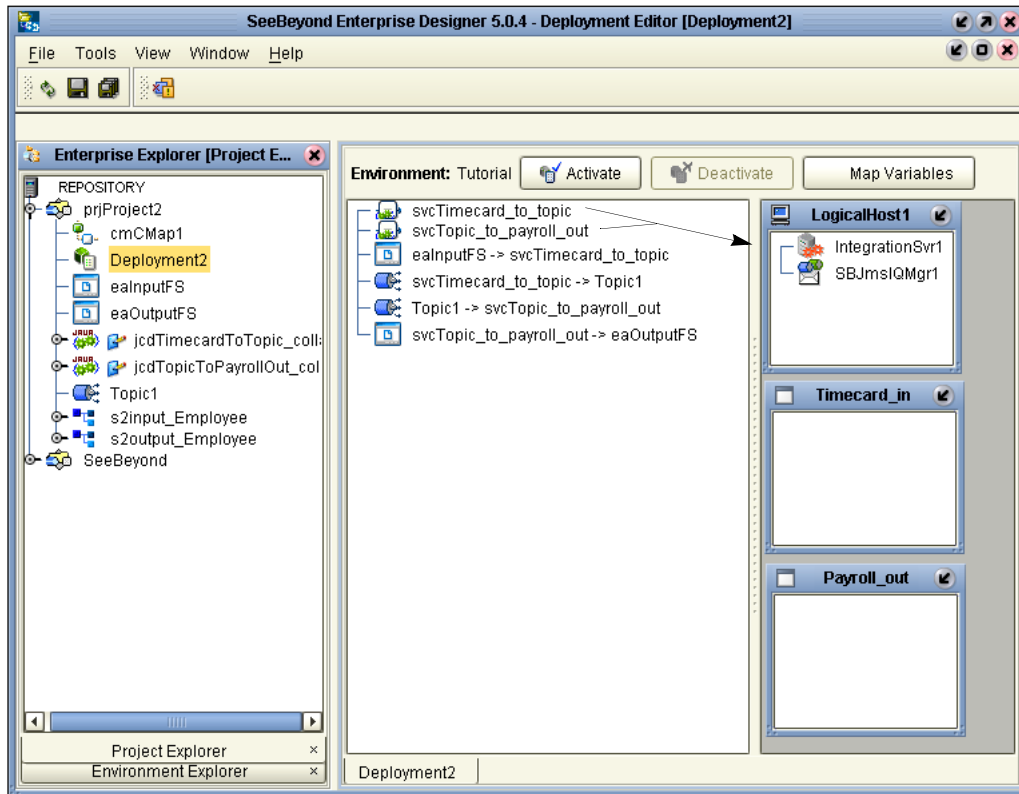
#### Deployment Profile

A Deployment Profile contains information about how Project components are deployed in an Environment, and it also maps components to the Environment.

- 1 Click the Project Explorer tab to return to the Project Explorer pane of your Project.
- 2 Right-click **Project2**, then click **New** and **Deployment Profile**.

- 3 Use the name **Deployment2** (and Tutorial).
- 4 Click **OK**.

**Figure 110** Deploy Components



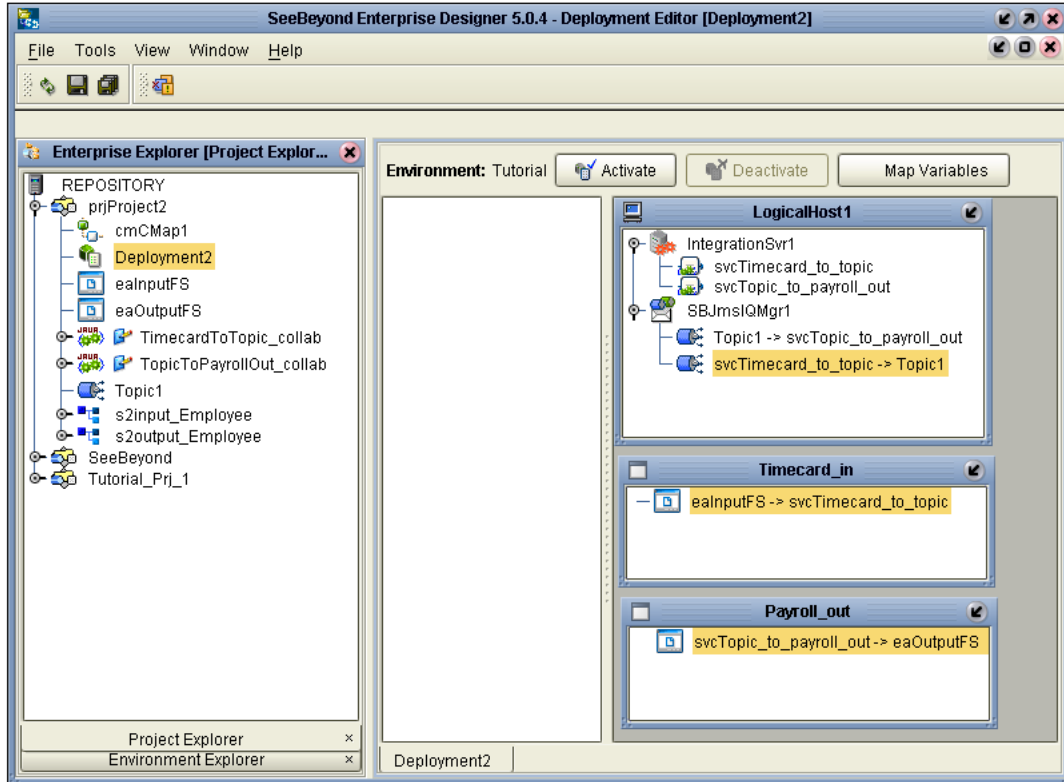
After creating your deployment profile, drag and drop components into the server dialog boxes.

### Drag and Drop Collaborations

- 1 Drag-and-drop your Services, **svcTimecard\_to\_topic** and **svcTopic\_to\_payroll\_out**, into your Integration Server, **IntegrationSvr1**. (See the previous Figure 110.)
- 2 Drag-and-drop your Topics into your Message Server, **SBJmsIQMgr1**.
- 3 Drag-and-drop your external files into your External File Servers, as shown in the following figure.

**Note:** If you are using the same Environment you used for Scenario1 (Tutorial), be sure to place the External files in the correct deployment boxes. For Scenario2 the **eaInputFS** goes into **Timecard\_in** and **eaOutputFS** goes into **Payroll\_out**.

Figure 111 Populated Environment



After populating your Environment with all the available components, there are no components listed in the middle pane. All your components reside in the servers, as shown in Figure 111 above.

---

## 4.15 Run Bootstrap and Management Agent

Refer to [Chapter 3, Run the Bootstrap and Management Agent](#) on page 49 in Scenario1.

---

## 4.16 Run Project2

**Note:** Refer to [Chapter 3, Run Project1](#) on page 51 in Scenario1, except this will be for Project2. If you have already ran Project1 in the first scenario, you should deactivate the deployment profile in Project1 before running Project2.

**Note:** You can use the same Environment (Tutorial) and same logicalhost that you used in Scenario1 for Scenario2, but the external file names must be different and the file

*names in each project must be different. Also, the deployment profile (name) must be different, Deployment2, for example.*

---

## 4.17 Verify Output Data

In “**Sample Data**” on page 60, you created a sample input XML file and set up the DTDs. The Management Agent processes your input file, **Timecard.xml**, and writes the results to **s2output1.dat**.

### View Output

- 1 Use Windows Explorer to navigate to **c:\eGateData\Project2**.
- 2 See that the Input eWay, after processing the input file, renamed the input file to **Timecard.~in**.
- 3 See that the Output eWay generated the output file, **s2output1.dat**.
- 4 Use Windows Wordpad or a similar text edit application to view the contents of the output file.

**Figure 112** Output File

```
<?xml version="1.0" encoding="UTF-8"?>
<Employee>
  <FName>Jane</FName>
  <LName>Smith</LName>
  <EmpNo>100</EmpNo>
  <Amt>2000 </Amt>
</Employee>
```

In this sample, the amount is 2000 because hours worked was 40 and rate of pay was 50. See input file, Figure 113.

**Figure 113** Input File

```
<?xml version="1.0" encoding="UTF-8" ?>
<Employee>
  <EmployeeNumber>100</EmployeeNumber>
  <LastName>Smith</LastName>
  <FirstName>Jane</FirstName>
  <JobTitle>Manager</JobTitle>
  <HoursWorked>40</HoursWorked>
  <Rate>50</Rate>
</Employee>
```

To process the input file again, rename **Timecard.~in** to **Timecard.xml**. The File eWay picks up the input file, **Timecard.xml**, and writes the results to **s2output1.dat**

or **s2output%d.dat** (where “d” is incremented with each write), depending on your Multiple Records per File setting (see Figure 104).

# Enterprise Manager Overview

The Enterprise Manager is a powerful monitoring tool that is useful for troubleshooting problems and viewing system activity at run-time. The Enterprise Manager (although not required for the tutorials) provides, alerts, error handling, logs, and graphic views (such as views of the Connectivity map) showing active and inactive Services.

For more detailed information about the Enterprise Manager see the *eGate Integrator System Administrator Guide*.

---

## 5.1 Browser-based monitor console

The Enterprise Manager is a browser based console that allows you to view and monitor projects dynamically, in real time.

- Monitor, start, and stop Services
- Monitor JMS topics and queries
- View and edit messages
- Monitor system logs
- Monitor and manage system alerts
- Monitor eInsight Business Processes (if licensed)

### 5.1.1 When to Use the Monitor

Enterprise Manager will notify you by flashing the affected component node structure when:

- There is a problem within the Project
- You have stopped a server within a Project or Environment, or stopped an object in a Connectivity map

### 5.1.2 Using Project View

Project view allows you to perform run-time monitoring and management of Deployments and Connectivity Maps.

### 5.1.3 Using Environment View

Environment view allows you to perform run-time monitoring and management of eGate and eInsight System components.

### 5.1.4 Locating the Problem Node

When a problem occurs, the Project node is decorated with a flashing highlight, which is propagated upward from the problem node. When you expand the tree, the problem node is highlighted and has a flashing exclamation point appended to it. The exclamation point association visually calls out the target node (or nodes) for troubleshooting the problem.

---

## 5.2 Starting and Using Enterprise Manager

### Start Enterprise Manager

Use Internet Explorer to point to the URL used by your installation to access the Enterprise Manager. Refer to the *SeeBeyond ICAN Suite Installation Guide*.

- 1 Start Enterprise Manager and type your username and password.
- 2 Click **Login**.
- 3 Click the **Home** tab.
- 4 Double-click **Monitor**.

You can click the Project tab or the Environment tab to monitor the state of the Logical Host(s) and the Project components deployed in them. You can also view the logs and change the log levels of the Logical Host.

It is beyond the scope of this tutorial to explain all the capabilities of the Enterprise Manager. The following examples demonstrate a few of its capabilities.

### 5.2.1 Monitor a Collaboration in your Connectivity Map

#### Enterprise Manager Graphic Tab

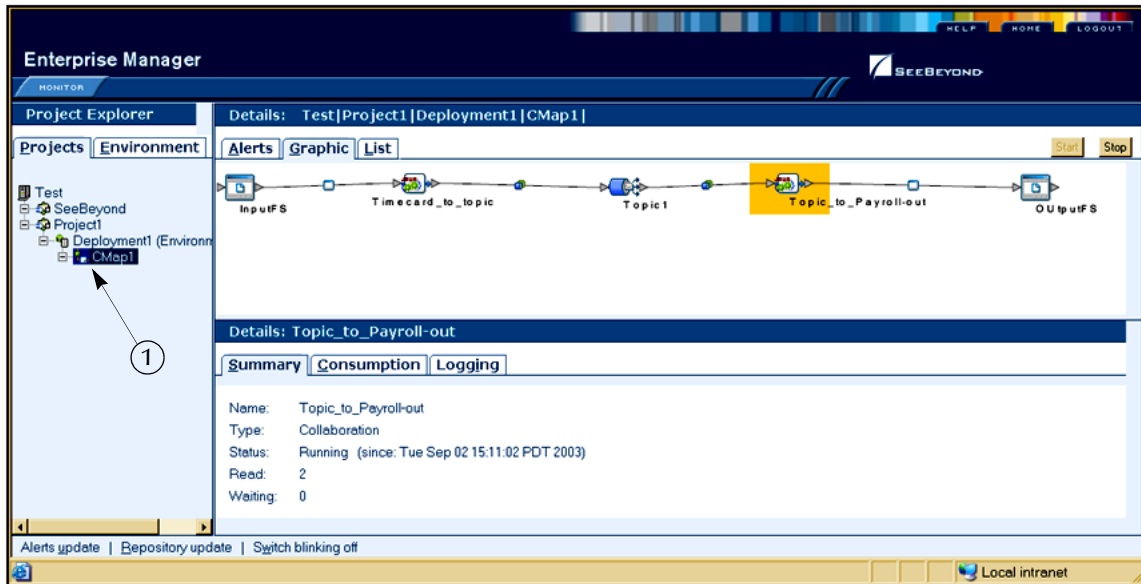
With the Enterprise Manager open, perform the following steps:

- 1 Click the Project tab.
- 2 Click one of the tabs to display information.
  - ♦ **List** shows deployment status
  - ♦ **Graphic** shows the Connectivity Map
  - ♦ **Alerts** shows message types and severity of the message
- 3 Click the Graphic tab. (If this tab is not available, you may need to download and install the SVG Viewer.)



- 4 Expand all the views by double-clicking the repository name (Test).
- 5 Click the CMap1 icon (number 1 in Figure 114).  
(The Connectivity Map appears.)
- 6 Double-click **Topic\_to\_payroll\_out**.  
Refer to the following Figure 114.

**Figure 114** Enterprise Manager CMap1



The Enterprise Manager shows that the collaboration Service is open for **Topic\_to\_payroll\_out**. The status shows that the application is running and there have been two reads of the file.

## 5.2.2 Stop Service

Using the example above, illustrated in Figure 114, you can also start and stop services. For example, if you want to stop the **Topic\_to\_payroll\_out** Service (stop processing the messages coming from Topic1 and instead hold them in a queue) you would perform the following steps:

### Stop Topic\_to\_payroll\_out Service

- 1 Click the **Topic\_to\_payroll\_out** Service (icon).

**Figure 115** Stop Service Button



- 2 Click the **Stop** button.

A red disk appears over the icon to indicate that the Service is stopped.

**Figure 116** Stop Service Indicator



Keep in mind that although the **Topic\_to\_payroll\_out** Service does not process messages after it is stopped, those messages are held in a queue to be processed immediately when the Service is restarted, by clicking on the Start button.

### 5.2.3 Monitor Alerts

#### Enterprise Manager Project Tab

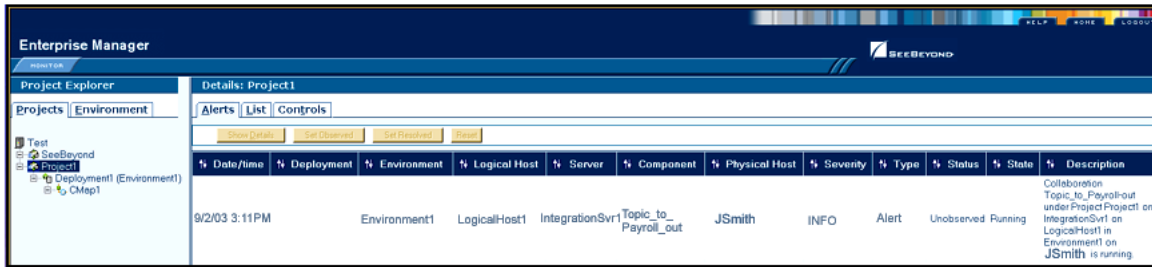
With the Enterprise Manager open and **Topic\_to\_payroll\_out** stopped (causes alert), perform the following steps:

- 1 Click the Project tab.
- 2 Expand all the views by double-clicking the repository name (Test).
- 3 Click the Project name.
- 4 Click the Alerts tab.

(The alert messages display with a description in the far right column.)

Refer to the following Figure 117.

Figure 117 Enterprise Manager CMap1



## 5.2.4 Check Log Messages

You can troubleshoot errors by viewing the Logged messages.

In this tutorial, for example, the input timecard is designed for integers. The business rules are set up to handle integers only. If a data entry person wanted to correctly type a pay rate of \$42.50, he or she would have to input the integer as 4250, assuming two decimal places. (The decimal places would have to be reformatted through additional processing.)

If a data entry person incorrectly entered a pay rate as 42.50 (timecard input), the decimal would generate an error message and the output file would not be updated.

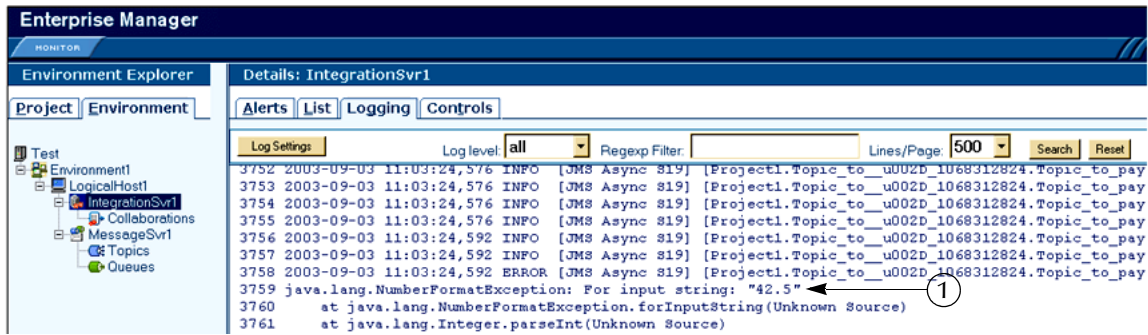
### View Logged Messages

With the Enterprise Manager open, perform the following steps to check logged messages:

- 1 Click **Environment**.
- 2 Expand all the views by double-clicking the repository name (Test).
- 3 Click **IntegrationSvr1**.
- 4 Click the **Logging** tab.
- 5 Click the **Search** button.
- 6 Scroll down to find error messages.

The error, **NumberFormatException** is recorded in the log (number 1 in the following Figure 118). Use the “search on page for” feature to find “exception.”

Figure 118 Check Logged Messages



# Glossary

## **Collaboration**

A logical operation performed between some combination of message destinations and external applications. The operation is defined by a Collaboration Definition, which can be encoded in either Java or XSLT.

Also see “**Service**” and “**Collaboration Definition**”.

## **Collaboration Definition**

The encoding of business rules, in Java or XSLT format. Typically, the encoding consists of operations on OTDs (see “**OTD**”). Several Collaborations can have the same Collaboration Definition.

## **Connection**

Consists of the configuration information that enables an eWay to connect to an external system.

## **Connectivity Map**

Contains business logic and routing information about the data transmission. A Connectivity Map usually includes one or more Collaborations, Passthrough Collaborations, topics, queues, and eWays. A Connectivity Map is created under a Project. A Project may have multiple Connectivity Maps.

## **Constants**

A name or value pair that is visible across a Project.

## **Deployment Profile**

Contains the information about how the Project components will be deployed in an Environment. A Project can have multiple Deployment Profiles, but only one Deployment Profile can be activated for a Project in any one Environment.

## **Derived Collaboration**

Collaboration that inherits operations from another, according to standard object-oriented practice.

## **eGate System**

See “**Project**”.

## **Environment**

A collection of physical resources and their configurations that are used to host Project components. An Environment contains logical hosts and external systems.

## **eWay**

A link between a Collaboration and an external connection including the message server connection (topic or queue) or external application.

**External Application**

A logical representation in an eGate Project of an external application.

**External System**

A representation in an eGate Project of an external application system.

**ICAN Suite**

The SeeBeyond Integrated Composite Application Network Suite.

**Integration Server**

J2EE software platform that houses the business logic container used to run Collaborations and JCA connectors (eWays). Provides transaction services, persistence, and external connectivity.

**JMS IQ Manager**

JMS-compliant, guaranteed delivery store, forwarding, and queueing service.

**Link**

The JMS Connection between a Collaboration and a topic or queue in a JMS-compliant message server.

**Linked Message Destination**

A reference to a Message Destination defined in another Connectivity Map.

**Logical Host**

An instance of the eGate runtime Environment that is installed on a machine. A Logical Host contains the software and other installed components that are required at runtime, such as application and message servers.

**Management Agent**

Uses J2EE technology to manage and monitor an eGate 5.0 deployment that may contain other application servers in addition to the SeeBeyond Integration Server. Defines management interfaces and services designed for distributed environments, focusing on providing functionality for managing networks, systems, and applications.

**Message Destination**

A general term for a topic or queue. Two or more Projects can share a message destination that has the same name and is deployed on the same message server. A single Project may also have a single message destination referenced in multiple Connectivity Maps.

**OTD**

An acronym for Object Type Definition. OTDs contain the data structure and rules that define an object. An OTD is used in Java Collaboration Definitions for creating data transformations and interfacing with external systems.

**Project**

Contains a collection of logical components, configurations, and files that are used to solve business problems. A Project organizes the files and packages and maintains the settings that comprise an eGate system in SeeBeyond's Enterprise Designer.

### **Queue**

A JMS queue is a shareable object that conforms to the *point-to-point* (p2p, or PTP) messaging domain, where one sender delivers a message to exactly one receiver. When the SeeBeyond JMS IQ Manager sends a message to a queue, it ensures it is received once and only once, even though there may be many receivers “listening” to the queue. This is equivalent to the subscriber pooling in other queue implementations. You can reference a queue that exists in another Connectivity Map or Project.

### **Repository**

Stores and manages the setup, component, and configuration information for eGate Projects. The Repository also provides monitoring services for Projects, which include version control and impact analysis.

### **Schema Runtime Environment**

An add-on in eGate 5.0 that provides the upgrade path for e\*Gate 4.x users to upgrade to eGate 5.0. Also known as the SRE.

### **Service**

Contains the information about executing a set of business rules. These business rules can be defined in a Java Collaboration Definition, XSLT Collaboration Definition, Business Process, eTL Definition, or other service. A Service also contains binding information for connecting to JMS Topics, Queues, eWays, and other services.

### **Subproject**

An independent Project that is included as part of another Project and listed on the Enterprise Explorer tree beneath the main Project icon.

### **Topic**

A JMS topic is a shareable object that conforms to the *publish-and-subscribe* (pub/sub) messaging domain, where one publisher broadcasts messages to potentially many subscribers. When the SeeBeyond JMS IQ Manager publishes a message on a topic, it ensures that all subscribers receive the message.

### **XSLT**

An acronym for Extensible Stylesheet Language Transformations. A file format used in eGate to generate Collaboration Definitions.

## e\*Gate 4.x Terms in eGate 5.0

Table 1 provides definitions for the terms that are new with eGate release 5.0, as well as equivalent terms from eGate release 4.x.

**Table 1** eGate 5.0 Terms

5.0 Term	4.x Equivalent Term
Collaboration	Collaboration
Collaboration Definition	Collaboration Definition
Connection	eWay Connection
Connectivity Map	Closest: Network View of an entire Schema
Deploy	Run the Control Broker
Deployment	<none>
Deployment Profile	Closest: Schema
Enterprise Designer	Enterprise Manager
Enterprise Manager	Enterprise Monitor
Environment	Schema (except only includes physical information, not business logic)
eWay	eWay Connection eWay
eWay Configuration	eWay Connection Configuration
External Application	eWay Connection
External System	eWay Connection
JMS Connection	eWay Connection
Integration Server	<none>
Link	JMS eWay Connection
Linked Message Destination	<none>
Logical Host	Participating Host
Message Destination	Topic or queue
Message Server	JMS IQ Manager
Object Type Definition (OTD)	Event Type Definition (ETD)
Process Manager	Control Broker
Project	Schema (except not including physical layer)
Queue	JMS queue
Repository	Registry
Subproject	Schema



**Table 1** eGate 5.0 Terms (Continued)

5.0 Term	4.x Equivalent Term
Topic	JMS topic
XSLT	<none>

# Index

## A

Adjust window size 85

## B

Bootstrap

The bootstrap process 49

Bootstrap Command

Example 51

Business Rules

Apply 76

Business Rules Designer window 76

Business Rules pane (see note) 83

Multiplication logic 81

Objects - in the Business Rules Designer window  
(see note) 77

## C

Call Java Method 84

Call New Constructor 86

Class Name box 85

Collaboration 109, 112

derived 109

Collaboration definition 109, 112

Collaboration Editor (Java)

Defined in a project 11

Collaboration Editor (XSLT)

Defined in a project 11

Connection 112

connection 109, 112

Connectivity Map 109, 112

Add components 22, 65

Create 20, 63

Default name, CMap1 21

Defined in a project 11

Graphic 13, 57

Graphic, unlinked 40, 90

Link components 40, 90

constants 109

Control Broker 112

conventions

path name separator 7

Windows 7

## D

deploy 112

Deployment 112

Deployment Profile 109, 112

Deployment Profile (Scenario1) 44

Deployment Profile (Scenario2) 96

Deployment Profile Editor

Defined in a project 11

derived Collaboration 109

document

conventions 7

## E

eGate system 109

Enterprise Designer 112

Create a new project 19, 61

Create and configure components 9

Deploy your project 51

Editor 11

Features 9

GUI 10

Menu Bar 10

Project 11

Start the Enterprise Designer 19, 62

Enterprise Explorer

Create a connectivity map 21, 64

Create and name a project 19, 62

Organize components, in the left pane 11

The left pane of the Enterprise Designer 10

Tree 81

Enterprise Manager 112

Check log messenger 107

Flashing indicators 103

Monitor a Collaboration 104

Monitor Alerts 106

Overview 103

Problem Node 104

Starting and using 104

Stop a Service 105

Enterprise Monitor 112

Environment 16, 109, 112

Activate 52

Create environment 44, 96

Defined in a project 11

Deployment profile 47, 98

in Enterprise Manager 104

LogicalHost1 45, 97

Message server 99

New File External System 46, 97–98

Populate 48, 99

Environment Explorer

Accessed from View menu 10

Collection of logical hosts 11

Create environment 44, 96

error messages 71

ETD 112  
Event Type Definition 112  
eWay 109, 112  
    Configure 41, 92  
    Graphic 42, 93  
    Input eWay 92  
    Link between External Application and a Service 40, 90  
    Output eWay 92  
eWay Configuration 112  
eWay Connection 112  
eWay Connection Configuration 112  
external  
    application 110, 112  
    system 110, 112  
External Applications 22, 65

## F

FileClient  
    Web service (in the Tutorial used to receive or write data) 29

## G

GUI  
    Graphical User Interface 6

## I

ICAN Suite 110  
Input  
    Inbound eWay 13, 57  
    Input file 54, 101  
    Input.dtd, create input file 18  
    InputFS 57  
    s2Input.dtd, create input file 60  
Integer 85  
Integration Server 110, 112  
Introduction 6, 9

## J

JMS  
    connection 112  
    e\*Way Connection 112  
    IQ Manager 112  
    JMS Client Configuration 95  
    queue 112  
    topic 113  
JMS IQ Manager 110

## L

link 110, 112  
linked message destination 110, 112  
Logical Host 96, 110, 112  
Logicalhost 16  
Login  
    Login menu 19, 62

## M

Management Agent 110  
Menu Bar 10  
message  
    destination 110, 112  
    server 112  
Multiple records 44, 95  
Multiplication Operation  
    Graphic 87  
    Method box 85  
    Operator 85

## N

network view 112  
Nodes  
    Connect 83  
    nodes 83

## O

Object Type Definition 24, 67, 110, 112  
OTD 110, 112  
    Create OTD 24  
OTD Editor  
    Defined in a project 11  
Output  
    Outbound eWay 13, 58  
    Output file 53, 101  
    Output.dtd, create output file 18  
    OutputFS 13, 58  
    s2output.dtd, create output file 61

## P

Participating Host 112  
Polling interval 43, 94  
Process Manager 112  
Project 110, 112  
    Bootstrap command 51  
    Configure input 28, 72  
    Configure output 77  
    Create 19, 61  
    Create and name 19, 62

## Index

- Create OTD 24, 67
- Create path to data 42, 93
- Default name, Project1 20, 63
- Deploy and activate 51
- Deployment 44, 96
- Deployment profile 47, 98
- Drop down arrow 80
- Graphic, showing components 40, 90
- in Enterprise Manager 103
- Overview 13, 57
- Verify output 53, 101

Project Diagram 15, 59

Project Editor

- Part of Enterprise Designer 11

Project Explorer

- definition 11
- pane (left side of window) 27

Properties configuration 43, 94

## Q

queue 111–112

## R

Registry 112

Repository 16, 111–112

Rules 76

Run Tester icon 71

## S

Save

- Save 26, 70
- Save All 26, 70

Scenario1 (Building a Project) 12

Scenario2 (Building a Project) 55

Schema 112

Schema Runtime Environment 111

Security Server 111

Service 28, 72

- file-read service 29, 73
- Web service 30, 74
- wizards 28, 72
- Write 74

SRE 111

Status tab 71

subproject 111–112

supporting documents 6

## T

Time Card system 18, 60

Timecard\_to\_topic 13, 57

Topic

- Create a Topic 66
- Topic1 58
- topic 111–113
- Topic\_to\_Payroll\_out 58

## U

Unmarshal Text 82

## W

Windows 17, 60

- The Bootstrap command is case sensitive on Windows 50
- tutorial runs on a Windows system 6
- Windows Explorer 17, 60
- writing conventions 7

## X

XML

- data 92
- file 41, 92

XSLT 111, 113