SeeBeyond ICAN Suite

eInsight Enterprise Service Bus User's Guide

Release 5.0

S E E B E Y O N D®

Implementation   59

Method Palette   79

List of Figures

List of Tables

Chapter 1

Introduction

The *eInsight Business Process Manager User's Guide* provides instructions and background information for all users of the SeeBeyond Technology Corporation™ (SeeBeyond™) eInsight Business Process Manager™ (eInsight) application. This chapter introduces you to this guide, its general purpose and scope, and its organization. It also provides sources of related documentation and information.

In this chapter

- **Overview** on page 10
- **Contents of This Guide** on page 11
- **Writing Conventions** on page 12
- **Supporting Documents** on page 12

1.1 Overview

This guide explains how to use eInsight. This user's guide includes information on the following topics:

- Installing eInsight
- Working with eInsight modeling elements and the Business Process Designer
- Creating and configuring business process models
- Upgrading eInsight

1.2 Intended Audience

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

1.3 Contents of This Guide

This document includes the following information:

- **Chapter 1, "Introduction"** provides an overview of this document's purpose, contents, writing conventions, and supported documents.

- **Chapter 2**, **"Introduction to eInsight Enterprise Service Bus"** provides an overview of the product.

- **Chapter 3**, **"Installation"** guides you through the installation of eInsight.

- **Chapter 3**, **"Designing Business Process Models"** describes business process elements and procedures related to building a business process model.

- **Chapter 4**, **"Configuring Business Process Models"** discusses eInsight configurations and options.

- **Chapter 6**, **"Exception Handling"** describes how to set up error handling within a business process.

- **Chapter 5**, **"Deploy and Monitor Business Process Models"** discusses deployment of the business process model.

- **Chapter 8**, **"Persistence and Monitoring"** provides instructions for setting up persistence and monitoring.

- **Chapter 6**, **"Implementation"** gives a step-by-step example of a simple eInsight implementation.

- **Chapter 8**, **"Upgrading eInsight"** gives an overview of the upgrade procedure for previous versions of eInsight.

- **Appendix A**, **"Method Palette"** describes the methods available from the eInsight Business Rules designer.

- The **"Glossary"** defines eInsight specific terms.

- **"Index"**

1.4 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.	**Bold** text	▪ Click **OK** to save and close. ▪ From the **File** menu, select **Exit**. ▪ Select the **logicalhost.exe** file. ▪ Enter the **timeout** value. ▪ Use the **getClassName()** method. ▪ Configure the **Inbound** File eWay.
Command line arguments and code samples	`Fixed` font. Variables are shown in ***bold italic***.	`bootstrap -p `***`password`***
Hypertext links	**Blue** text	For more information, see **"Writing Conventions" on page 12**.

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.

1.5 Supporting Documents

The following SeeBeyond documents provide additional information about the SeeBeyond Integrated Composite Application Network Suite:

▪ *SeeBeyond Integrated Composite Application Network Suite Primer*

▪ *eGate Integrator Installation Guide*

▪ *eGate Integrator User's Guide*

Introduction to eInsight Enterprise Service Bus

This chapter provides an introduction to the eInsight Enterprise Service Bus (eInsight ESB) application.

2.1 Overview of eInsight ESB

eInsight Enterprise Service Bus (eInsight ESB) is a stand-alone application integration and business process management solution. eInsight ESB provides a clear view into the internal and external processes of your organization.

Business process management is a strategic orchestration of the movement of information and the flow of complex processes between participants (systems, users, and organizations) to accomplish larger business objectives.

2.1.1. SeeBeyond ICAN Suite Integration

eInsight ESB is tightly integrated with the ICAN Suite and is compatible with most other ICAN Suite products, including:

- eVision Studio
- eView Studio
- eIndex Global Identifier
- ePortal Composer
- eBAM Studio
- eTL Integrator
- Many of the ICAN eWays

2.1.2. eInsight ESB Architecture

The technologies that carry out eInsight ESB business process functions are entirely based on industry standards. eInsight ESB uses the following standards:

- Web Services Business Process Execution Language (**WSBPEL**) is the underlying code generated when creating a business process.

- Web Services Description Language (**WSDL**) is an XML-based language used to define Web services and describe how to access them.

- Business Process Markup Notation (**BPMN**), from the Business Process Management Initiative (BPMI) standards body, provides a standard graphical view for Business Process Execution Language for Web Services (**WSBPEL**).

- J2EE Connector Architecture (**JCA**) provides a mechanism to access external applications and data. The JCA engine is implemented as a standard JCA 1.5 module and it plugs into the SeeBeyond Integration Server.

2.1.3. Business Process Modeling and Design

You can use eInsight to streamline operations by creating business logic that helps you reach outward to include customers and trading partners. Using eInsight ESB to implement business process management removes inefficiencies by orchestrating a unified work flow. This flow can include multiple systems/users, therefore extending to customers.

eInsight ESB allows you to model the business processes that your department or even your entire company performs on a regular basis. The tools provided allow for various scenarios and events that may take place in your process.

2.1.4. Business Process Designer

The Business Process Designer serves as the front-end design tool used to create a visualization of your business workflow and increase understanding of the business processes involved. As the business user, you are able to integrate logic into the business process. The Business Process Designer includes the following major areas:

- **Enterprise Explorer -** Displays a hierarchical representation of all the business process models and related components. This view shows you what is currently being displayed on the modeling canvas.

- **Business Process Designer -** Used to graphically create the business process model in the form of an activity diagram.

- **Business Rule Designer** - Allows you to visually add business logic and to transform and map data.

2.1.5. Design Phase Overview

The basic steps that you will perform to design a business process model are as follows:

1 Plan and design a model that represents a business process taking place in your company.

2 Set up an Project and related components necessary to your business process model.

3   Create the new business process model in the eInsight Business Process Designer, using activities, links, decision and exception handling logic, and any other elements that express the actual business process.

4   Validate, generate and save the business process code to the Repository, where business processes configuration and deployment information is stored.

5   Create a Connectivity Map to configure the relationship between your components.

6   Select or create an Environment where your business process will run.

7   Select or create a Deployment Profile and activate it to complete the deployment process.

2.1.6. Runtime Phase Overview

Once all of the design phase tasks are complete and the system is running, you can monitor and manage business process activity and the overall Project with the Enterprise Manager.

In the runtime phase, processes are coordinated that enable the execution of business processes, activities and tasks. Runtime phase tasks:

- Receive messages that instantiate business process instances.
- Write monitoring, persistence and recoverability data to a database.

eInsight ESB Database

eInsight ESB components connect to a database to provide:

- **Persistence** - Instance data is written to a database table to ensure that data is able to persist in the system.
- **Recoverability** - Using the database allows you to recover data from the last state of consistency.
- **Monitoring** - Instance data is written to database tables and then read by the Enterprise Manager to provide current and historical system information.

2.1.7. Monitoring

Monitoring is available from the Enterprise Manager. This interface allows you to monitor and fix some problems that occur while the business process is running. Some actions that you can take from the Enterprise Manager include:

- Start, stop, restart, and pause components.
- Fix errors originating from erroneous data.

You can ensure that processes continue to run and work properly with these powerful monitoring tools.

2.1.8. **eInsight ESB Features**

eInsight ESB provides the following additional features:

- **Resource Management** - eInsight ESB uses a distributed and open architecture that enables components to access system resources (memory and processing power) as needed and in conjunction with other components.

- **Security** - The security module fulfills security needs such as authentication and authorization access to eInsight ESB functions.

- **Repository storage and access** - The setup, component, and configuration information for the elements of a Project, including business process and related eInsight ESB components, are stored in the Repository.

- **Deployment abilities** - Deployment profiles contain the information necessary to activate a Project business processes and associated components. When a deployment profile is activated, active business processes are made available as Web Services and published to SeeBeyond's UDDI Registry.

- **Monitoring** - The Enterprise Manager provides web-based monitoring abilities to observe and correct business process activity.

- **Connectivity Mapping** - The Connectivity Map maintains the relationships between system components.

- **Version Control** - This feature allows you to maintain multiple versions of the same components and business processes, through a check in and check out process.

- **Impact Analysis** - Impact Analysis allows you to view how changes to one component or business process will impact other components or business processes of a Project or all Projects in the Repository.

- **Import and Export of Business Process Models** - The ability to import and export business process models makes it possible to recreate the processes on other systems or to reuse processes that may be similar in other areas.

<div align="right">

Chapter 3

</div>

Designing Business Process Models

You can use the Business Process Designer to configure the components depicted by each activity in your business process models. This chapter provides the background information you need to create and understand business process models.

3.1 Overview

Topics in this chapter are:

- **"Building a Business Process Model" on page 17**
- **"Using the Business Process Designer" on page 19**
- **"Modeling Elements" on page 20**

3.2 Building a Business Process Model

A *business process* is a collection of actions that take place in your company, revolving around a specific business practice. These processes can involve a variety of participants and may include internal and external computer systems or employees. In the Business Process Designer, you create a graphical representation of the business process called a *business process model*.

A business process modeled in the Business Process Designer may look something like Figure 1.

Figure 1 Sample Business Process Model
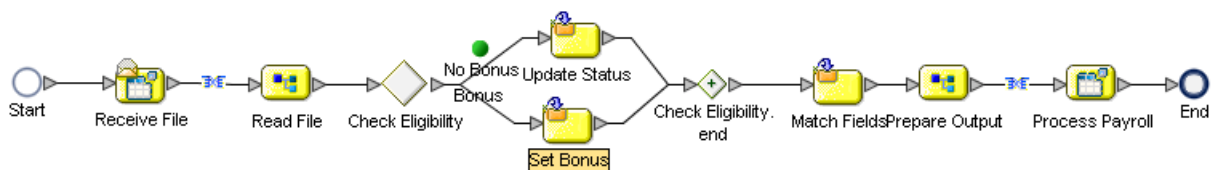
Add a Business Process to your Project

Adding a business process to your Project provides an empty modeling canvas for you to work with and allows you to name the business process. Before you create a model of your business process, you must add a new business process to your project.

1   From the Enterprise Explorer pane, right-click on a **Project**.

2   Select **New** and **Business Process** from the context menu.

3   Enter a new name for your Business Process.

Note: *See the eGate Integrator User's Guide for more information about creating a new Project.*

3.2.1. Modeling a Business Process

To model a business process, drag and drop modeling elements on the Business Process Designer, and then link these components to reflect the logical flow of the business process. The Business Process Designer provides the tools you need to quickly develop business process models, including graphic editing tools to help you adjust, size, and align model components.

Business Process Designer

Once you create a new business process, you will build your model in the Business Process Designer (as shown in Figure 2). The Business Process Designer is the area in the Enterprise Designer where you view, create, and edit your business process models.

You can also drag and drop many other ICAN component operations from the Project Explorer directly onto the Business Process Designer or an activity. Some of the other component operations that you can use in a business process include:

- File eWay: Read and Write operations
- Object Type Definition (OTD): Marshal and Unmarshal operations
- Java Collaboration Definition operations
- eVision Pages and PageFlows

Create a Business Process Model

Begin designing your business process model by dragging and dropping modeling elements onto the Business Process Designer from the toolbar.

The **Start** and **End** Activity appear on the blank Business Process Designer, by default. There is only one starting point for any business process model. There can be multiple end points.

1   Drag the appropriate modeling elements to your blank business process to the Business Process Designer (Figure 2).

Figure 2   Business Process Model

Project Explorer   Business Process Designer
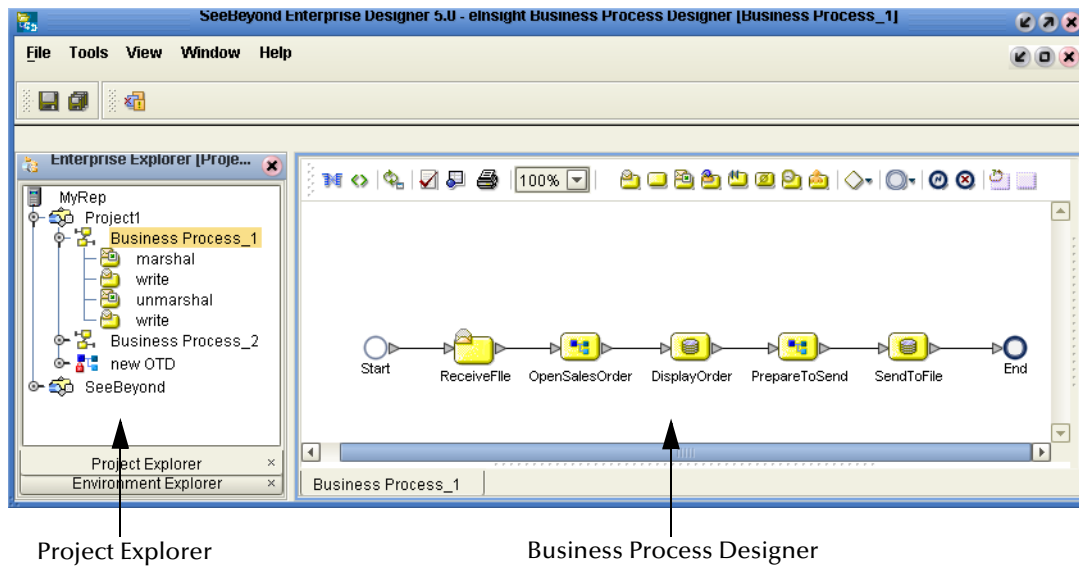
2   Draw links between the modeling elements to show the process flow (Figure 2)

3   Select **Save** to save your changes to the SeeBeyond Repository.

This will validate your business process, generate the code to run it and save your changes to the SeeBeyond Repository.

3.3   Using the Business Process Designer

Using the Business Process Designer is very similar to any of the other ICAN Suite interfaces. When you create a new Business Process, you see the Business Process Designer and the a new Business Process Designer toolbar appears, as shown in Figure 3.

Figure 3   Business Process Designer Toolbar Options
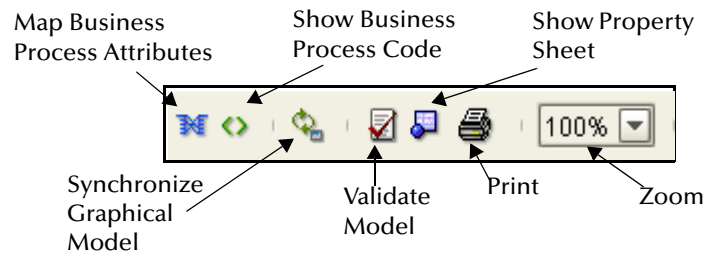
- **Map Business Process Attributes** - Selecting this icon reveals the Business Rules Designer in the lower portion of the Business Process Designer.

- **Show Business Process Code** - You can toggle this icon to see and edit the corresponding Business Process Execution Language (BPEL) code.

- **Synchronize Graphical Model and Business Process Code** - This icon will update the BPEL code on demand. The BPEL code is also synchronized when the model is saved.

- **Validate Business Process Model** - Click this icon to check for any errors in your Business Process Model.

- **Show Property Sheet** - This icon shows the Property Sheet for the modeling element that is selected.

- **Print** - You can print the model from the toolbar. This options also allows you to control the scale of the printed model.

- **Zoom** - Zoom control over the view of the model is available from the toolbar.

3.4 Modeling Elements

The Business Process Designer provides a palette of modeling elements to assist you in customizing your business process model. The Business Process Designer is where the you create the business process flow. Like other objects, business processes appear in the Enterprise Explorer.

Elements from the Enterprise Explorer can either be dropped onto empty canvas or onto an Activity. Many elements provide custom settings so that you can model every detail of your process. Each business process model you create consists of some or all of the elements as described in the following sections:

- **Activity Elements** on page 20
- **Branching Activities** on page 22
- **Intermediate Events** on page 23
- **Scope** on page 24
- **While** on page 24

Activity Elements

There are several different kinds of activities you can include in a business process model. Table 2 shows examples of each of the different kinds of activities described below.

Add an activity

1 To add an activity to the modeling canvas do the following:

2 Click an activity from the Business Process Designer toolbar or the Project Explorer list, then drag and drop it where you want it.

3 Click on the activity name and begin typing to rename it from the default. The activity name must contain at least one character (A-Z, a-z, or 0-9), it must start with a letter or an underscore (_) and it may contain spaces.

The selected activity appears on the modeling canvas.

Link modeling elements

The Business Process Designer supports orthogonal and diagonal link styles – this setting applies to all links in a model and is an automated application of the style.

To link modeling elements

1   Move your cursor over the connector portion of your modeling element.

2   Hold the cursor over the outside edge of the modeling element until it changes from the arrow pointer to a hand (see Figure 4).

Figure 4 Link Example
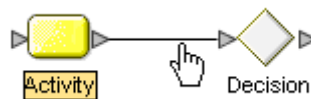
3   Drag a line from the first activity to the connector of the second activity, and then release the mouse.

Table 2 Activity Elements

Button	Command	Function
	Start Node	The Start Node is a modeling element indicating the start of the process. This element appears in the Business Process Designer, by default, when you create a new business process model. A Start Node can only link to a **Receive Activity**.
	Link	Links indicate the flow of the business process by connecting activities together. The Business Process Designer ensures the model is being properly linked because it does not allow invalid links to connect. Links can also accept business rules. A link with a business rule is marked with a blue icon.
	End Node	The modeling element indicating the completed state of a business process. This element appears in the Business Process Designer, by default, when you create a new business process model.
	Receive Activity	This Activity type is the only element that a **Start Node** can link to. It is used to indicate the invocation of a business process or to wait for the arrival of an inbound message. The Receive Activity represents the actual method by which a business process is initiated.
	Activity	An Activity is a step in the business process in which the The Business Process Designer Engine will invoke a Web Service or an ICAN component. Depending upon the configuration of the component, a response may or may not be required.

Table 2 Activity Elements

Button	Command	Function
	Reply Activity	The Reply Activity allows a business process to respond to the external system or user that originally invoked the business process. The original receive at the beginning of the business process is paired with the Reply at the end of the process. In cases where a message must be sent back to the caller of the business process, the Reply uses information that correlates the message in the calling system. A Reply acts as the last step in a business process in which the business process is acting as a Web Service or sub-process. A Reply correlates the outbound message back to the calling process, for example, it can reply to an external system as a Web Service.
	Business Rule Activity	This Activity is used to map and manipulate data in the Business Rules Designer. You can also add a Business Rule to some links for the same purpose.
	Compensate	This element is used to invoke compensation on an inner scope that has already completed normally. This construct can be invoked only from within a fault handler or another compensation handler.
	Empty Activity	This activity allows data to pass through without any changes.
	Wait Activity	The Wait activity acts as a timer. You can build a model in which there are two simultaneous paths within a set scope, one for the business process and one for the timer. If the timer condition takes place first, an exception is thrown, handled and the business process path is abandoned.
	User Activity	This activity is used when assigning, escalating or otherwise using human intervention to complete business process tasks.

Branching Activities

Branching Activities are objects you add to your business process models to specify the logical flow of information. There are three different kinds of Branching Activities—Decisions, Event Based Decisions, and Flow.

Add a Branching activity

To add a Branching Activity to the modeling canvas:

1 Click on the Branching Activities toolbar icon and select the type of Branching Activity you would like to add.

2 Click on your choice and drag it from the menu to the Business Process Designer canvas.

The selected Branching activity appears on the modeling canvas.

Table 3 Branching Activities

◇	Decision	A Decision allows one of several possible paths to execute, based on expression logic. This element is used to create complex expressions that determine the path of the business process. It also contains the expression and connection names. Decisions allow you to define expressions that are evaluated to determine the proper business process flow. Expressions are built using the Business Rules Designer interface and Business Process Attributes.
◈	Event Based Decision	Multiple possible messages can be juxtaposed against a timeout condition to allow the type of message received to determine the appropriate business process path.
◈	Flow	Allows you to specify one or more activities to be performed concurrently.

Intermediate Events

Intermediate Events are those activities that can interrupt a Business Process flow. Some intermediate events handle exceptions that may occur during your business process or compensate for exceptions that occur.

Add an Intermediate event

To add an **Intermediate event** to the modeling canvas:

1 Click on the **Intermediate event** toolbar icon and select the type of **Intermediate event** you would like to add.

2 Click on your choice and drag it from the menu to the Business Process Designer canvas.

Table 4 Intermediate Events

⊙	Timer Event	A Timer Event is set upon Activities, sets of Activities or a business process as a whole to ensure that process(es) complete within given amount of time. Timeout conditions also allow you to design the process that takes place after a timeout condition takes place.
◉	Message Event	This is similar to a Receive Activity, but it occurs only in the middle of a process. Each of these elements can be a different message.
⊙	Catch Named Exception	Each automated system (backend system) or Web service can publish their possible error codes (for instance, fault 15 is "bad data"). Those codes can be mapped to exception handlers. Each exception handler is connected to the scope that surrounds one or more steps in a business process. The components within that scope will throw the exceptions when things go wrong and the exception handler will automatically initiate the appropriate process to handle the problem.
⊙	Catch All Exceptions	This exception handler is configured to handle all exceptions that occur in a scope.

Table 4   Intermediate Events

	Compensation Handler	Used when something in a business process fails and requires a rollback or upstream activities (like money has to be returned to the customer). On an automatic basis in the business process, upstream steps in the business process are notified that the failure has occurred and certain transactions need to be reversed, sometimes in a sequential order. The compensation handler allows you to design the process and circumstances in which the compensation takes place.
	Throw	This element exists in case you want to create an error along a certain business process.
	Terminate	This element allows you to terminate a business process path, before it would normally end.

Scope

The behavior for one or more activities can be defined by a scope. A scope can provide exception handlers, event handlers, a compensation handler, and data variables. The exception handlers for the scope can be used to catch the faults caused by the possible exception responses.

Table 5   Scope Element

	Scope	The Scope element allows you to apply exception handling to a set of sequential or simultaneous steps in a business process.

While

This element makes it possible to have repeating or looping logic inside of a business process.

Table 6   While Element

	While	This allows you to create a looping process within a business process (for instance, a negotiation process may take several weeks, but the manager wants to review the daily status). The loop continues until the negotiation is complete, and then the business process continues.

3.4.1. Validating a Business Process Model

After creating a business process model, you can check to see if there are any problems such as activities that are not connected or an incorrect number of output links from an activity.

To check the business process for errors

- On the toolbar, select **Validate Business Process Model.**

  ◆ If an error is encountered, a message box displays information about the error.

  ◆ If there are multiple errors, you see a Next option to view each error.

◆ If there are no errors, a message appears stating that there were no errors**, as shown in**

Figure 5 Validate Business Process Model
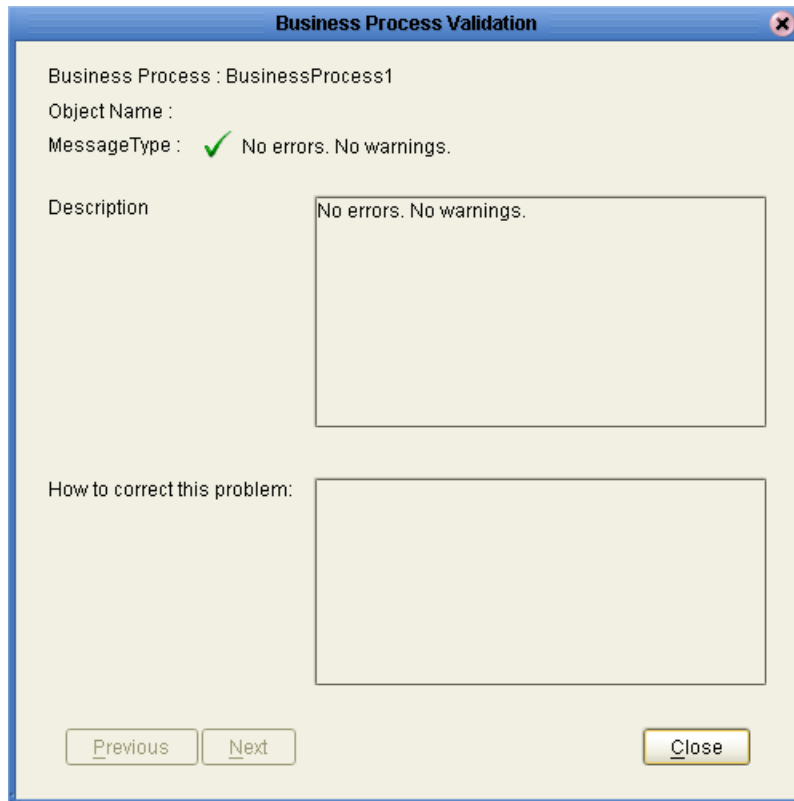
3.4.2. Saving a Business Process Model

Even if a business process model is not complete and/or contains errors, you can save it as a "work in progress" and return to it later.

To save a business process model

1   Do one of the following:

◆ Pull down the **File** menu and select **Save**

◆ Press **Ctrl+s** on the Keyboard

◆ On the toolbar, click **Save**.

<div align="right">

Chapter 4

</div>

Configuring Business Process Models

This chapter provides the background information you need to configure business process models.

4.1 Overview

Most of the advanced modeling elements and some of the basic modeling elements allow you to configure settings that customize your business processes. Topics in this chapter are:

- **Configuring Modeling Elements** on page 26
- **Business Process Properties** on page 33
- **Exception Handling** on page 46

4.2 Configuring Modeling Elements

Some modeling elements have configuration options. This section describes those elements and how to configure each option.

4.2.1. Business Rule Activity

The Business Rule Activity is used to map and manipulate data in the Business Rule Designer. You can also add a Business Rule to some links for the same purpose.

Business Rule Links

You can configure logic in a Business Rule Activity or add a Business Rule to a link.

Add a Business Rule Activity

1 Select the Business Rule Activity from the Business Process toolbar.

2 Drag the Business Rule Activity to the Business Process Designer.

3 Click the **Map Business Process Attributes** icon on the toolbar.

The **Business Rules Designer** appears in the lower part of the Business Process Designer.

Add a Business Rule to a Link

1 Right-click on a link that you have created.

2 Select **Add Business Rule**.

3 Click the **Map Business Process Attributes** icon on the toolbar.

The **Business Rules Designer** appears in the lower part of the Business Process Designer.

Business Rule Designer

The Business Rule Designer allows you to configure relationships between Input and Output Attributes. Some attributes are automatically configured for each activity when you drag and drop a component on the Business Process Designer (as shown in Figure 6).

The Business Rules Designer appears when you click the **Map Business Process Attributes** icon (as shown in Figure 6). It is active when you:

▪ Add or select a link with a Business Rule.

▪ Add or select a Business Rule activity.

Figure 6 Business Rule Designer

Method Palette

Business Rule Designer
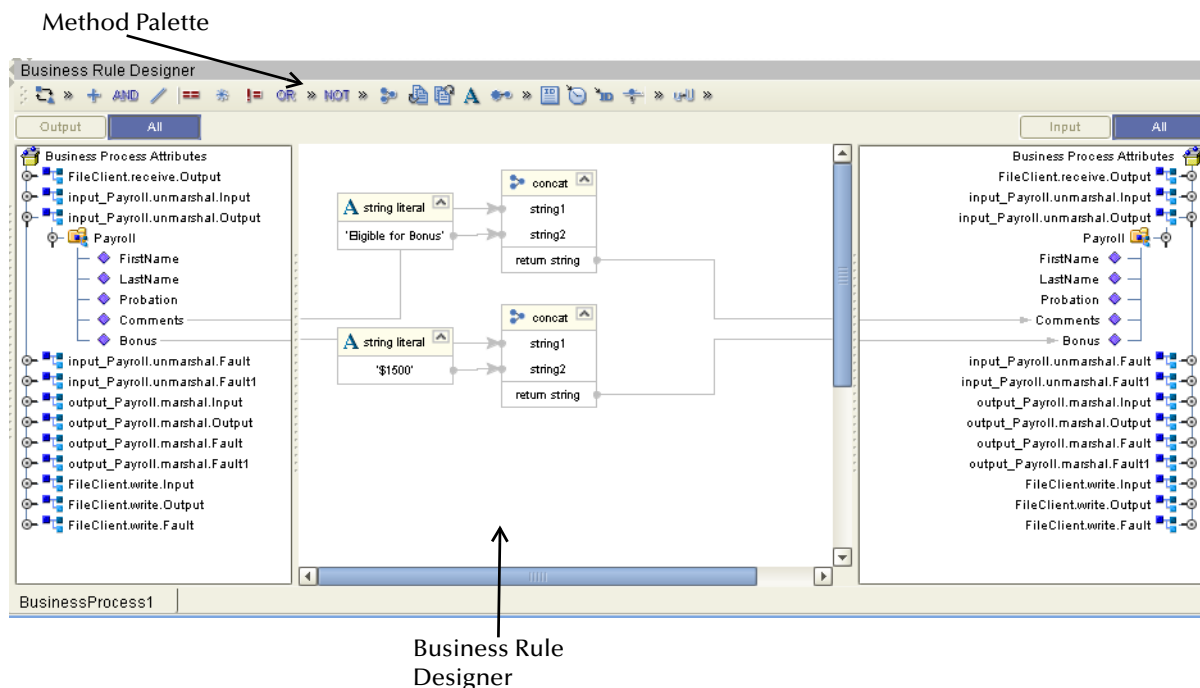
Method Palette

Use the Method Palette in the Business Rule Designer (as shown in Figure 6) to configure data passed between input and output nodes. You can drag and drop a method from the method palette to the Business Rules Designer and then configure the method.

See **"Method Palette" on page 79** for more information about each method available in the Method Palette.

4.2.2. User Activity

User activities allow you to access attributes in the business process. A User Activity requires human input to continue the business process. A set of eVision pages allow you to access attributes for the User Activity. You can then process and manipulate the attributes and send them back to eInsight ESB. eInsight ESB uses your input to continue the business process.

1 Create the Business Process and User Activity.

2 Set up a repository of users, organizational structures and roles in LDAP. (See **Organizational Roles** on page 29.)

3 Set up user assignments. See **Configure Task Assignment** on page 30.

4 Use the Worklist Manager to **Manage Tasks** on page 32.

User Activity Configuration Overview

This section provides an overview of how to configure the User Activity. This option uses LDAP to determine who belongs to the organizational structure. You can also set the options to escalate a task to the manager's queue, using time-out or another condition.

Configure a User Activity

The basic steps to configure the **User Activity** are as follows:

1 Create a business process model.

2 Configure your database for persistence and run the WLM Database Scripts.

3 Add a User Activity as part of your business model.

4 Create an eVision Studio PageFlow (see the eVision Studio User's Guide for details).

5 Drag the eVision PageFlow to the User Activity.

Figure 7   Configured User Activity

6 Double-click the User Activity and connect to an LDAP directory to look up users in your organization.

7 Select the task assignment, time-out and escalation options that you require.

Figure 8 Worklist Manager Settings
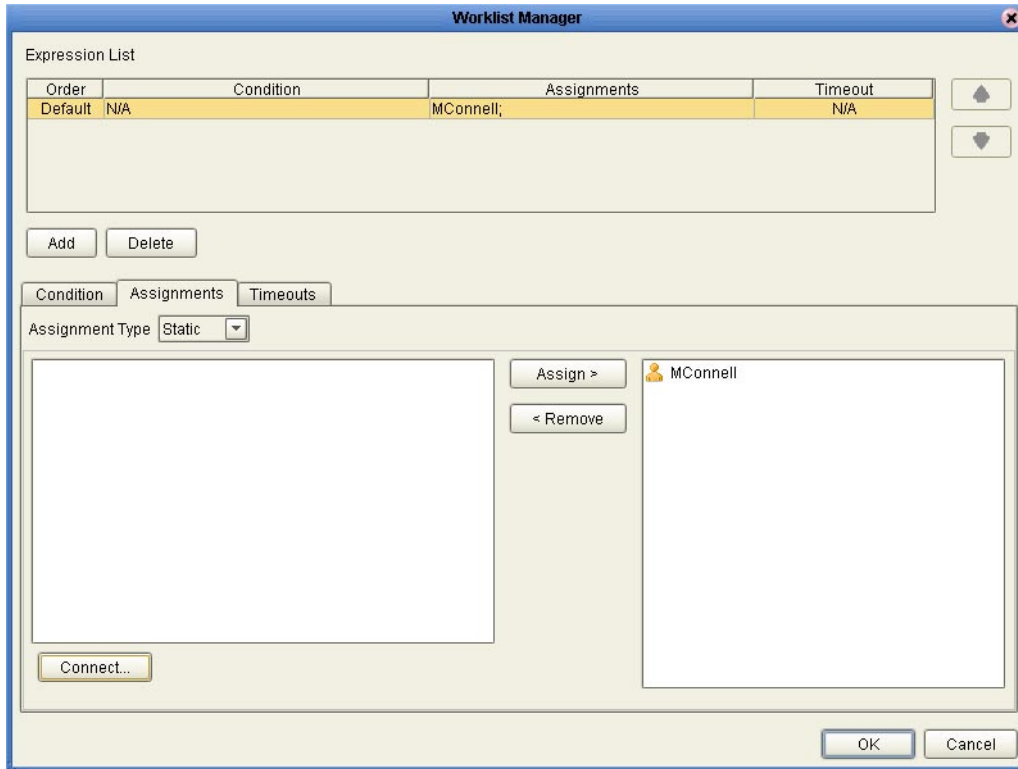
4.2.3. Organizational Roles

Organizational roles help define processes based on a person's position or title. By entering information about the structure of your organization, you can make processes easier to manage.

Support for LDAP means that you can use repositories of users, hierarchical organizational structures, and roles. An LDAP based applications can be used to populate the Worklist Manager with members of your organization and their organizational role. You assign rights based on organizational role assignments.

Figure 9   LDAP Connection Settings
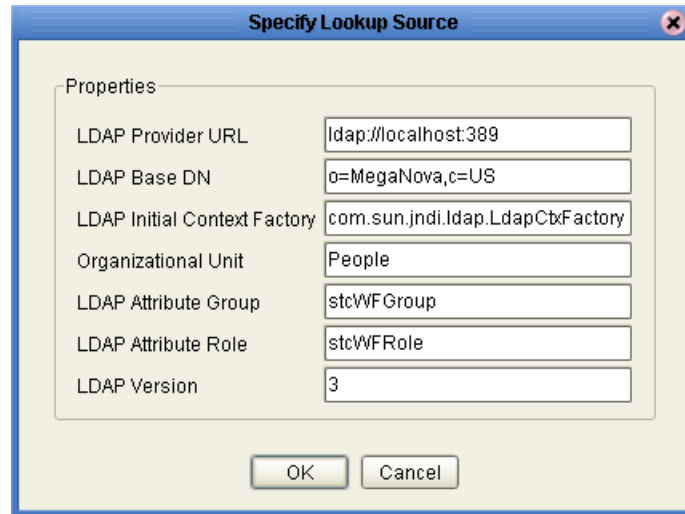
For example, you may allow managers to view their subordinate's activity list (also called a **Worklist**) and the ability to reassign tasks in that list. As a manager, when you login to your Worklist, you see your tasks and the tasks of your direct reports.

Note: See your third party software vendor's manual for information about setting up your organization's information in LDAP.

4.2.4. Task Assignment

Task assignment allows you to set up and view user queues, depending on your organizational role. With certain management level rights, you may assign a person who will receive a task, if the activity fails.

Configure Task Assignment

1   Double-click the **User Activity**.

The Worklist Manager dialog displays (see Figure 10).

2   Click **Connect** to find a look up source, as shown in Figure 9.

The user list displays (See Figure 10).

3   Click **OK** to return to the Worklist Manager dialog.

Figure 10 Worklist Manager: Assignments Tab
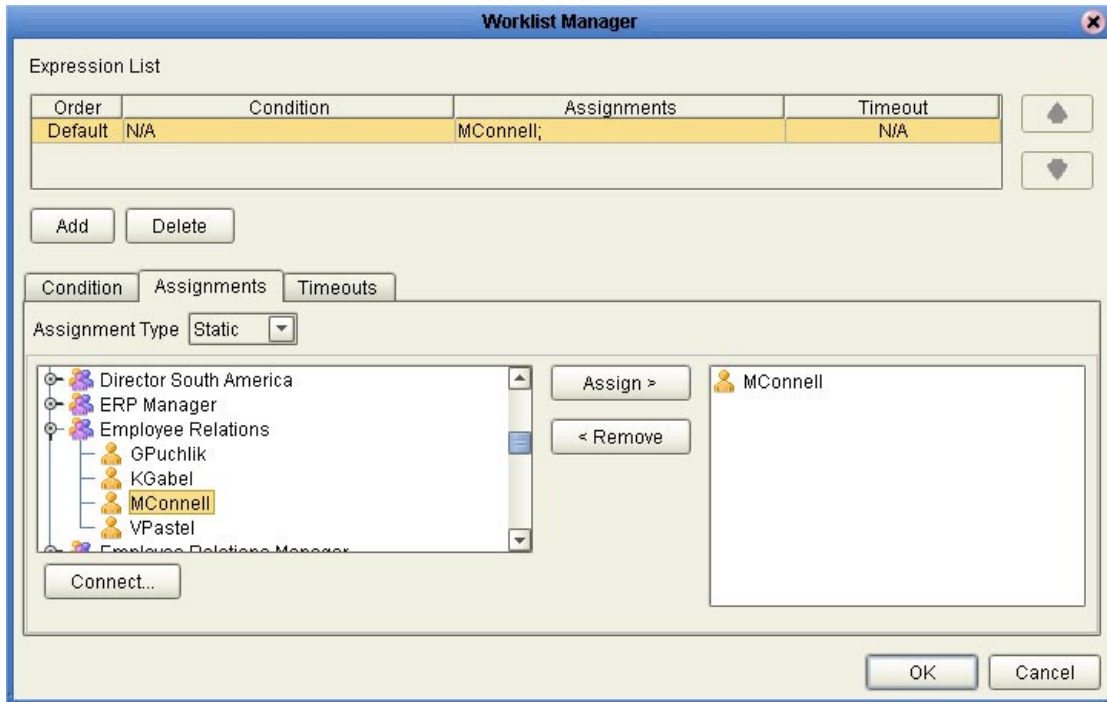
4   Navigate the Roles or Groups list to find your assignees**.**

5   Select individuals from the list and click **Assign**.

6   Select the **Condition** tab to enter an expression.

You can create expressions in the Business Rule environment to evaluate an activity for user completion.

7   Select Timeout and Escalation settings from the Timeouts tab.

8   You may also:

  ⬥ Add another rule by choosing **Add**.

  ⬥ Delete an assignment from this page by selecting an expression and choosing **Delete**.

9   Click **OK** when you have completed the configuration.

4.2.5. Using the Worklist Manager

Login to the Worklist Manager to view your list of tasks. You will see your own tasks as well as any subordinates assigned to you (if applicable). You can manage your tasks and/or the tasks of your subordinates from this view. See Figure 12 for an example of a manager's Worklist view.

Manage Tasks

From the Worklist Manager, you can manage your tasks and/or the tasks of your subordinates. You can access the Worklist Manager with a web browser by entering:

```
http://<localhost>:<port>/wlm
```

- <localhost>: The system where your Repository is running.

- <port>: The port number to access your Repository.

Figure 11 Worklist Manager Login
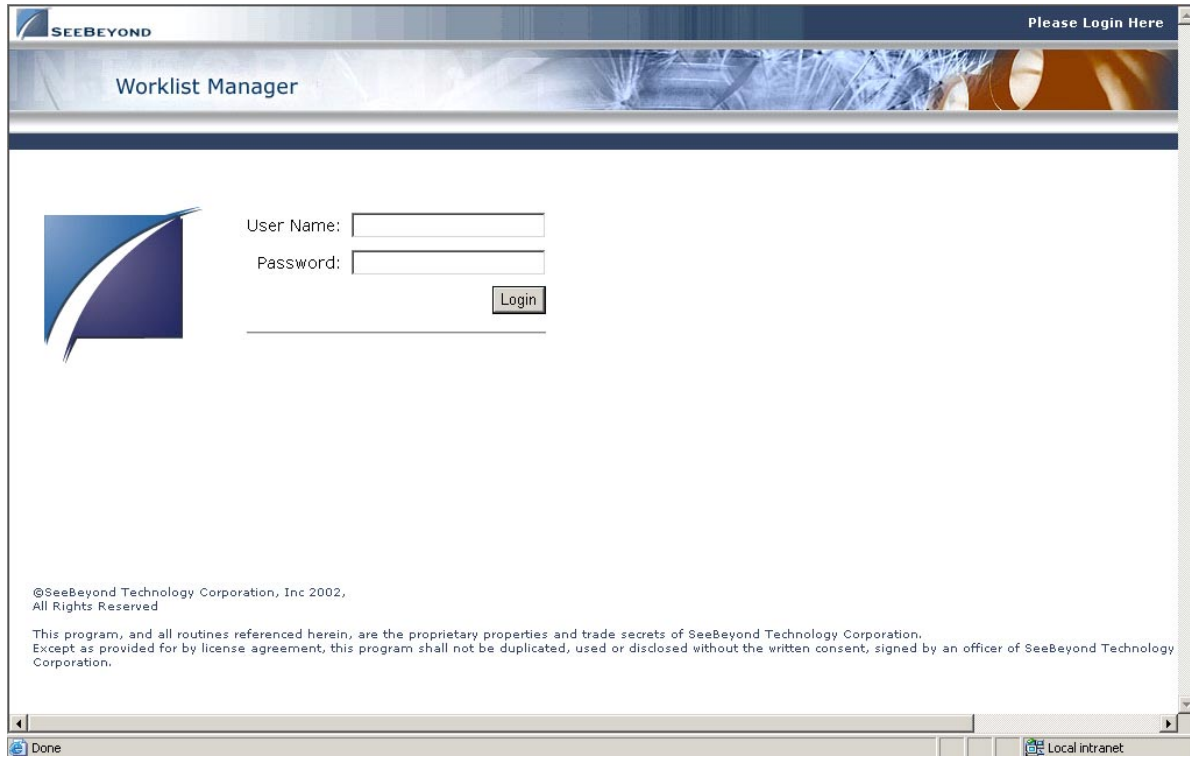
Figure 12 Worklist Manager: Task View
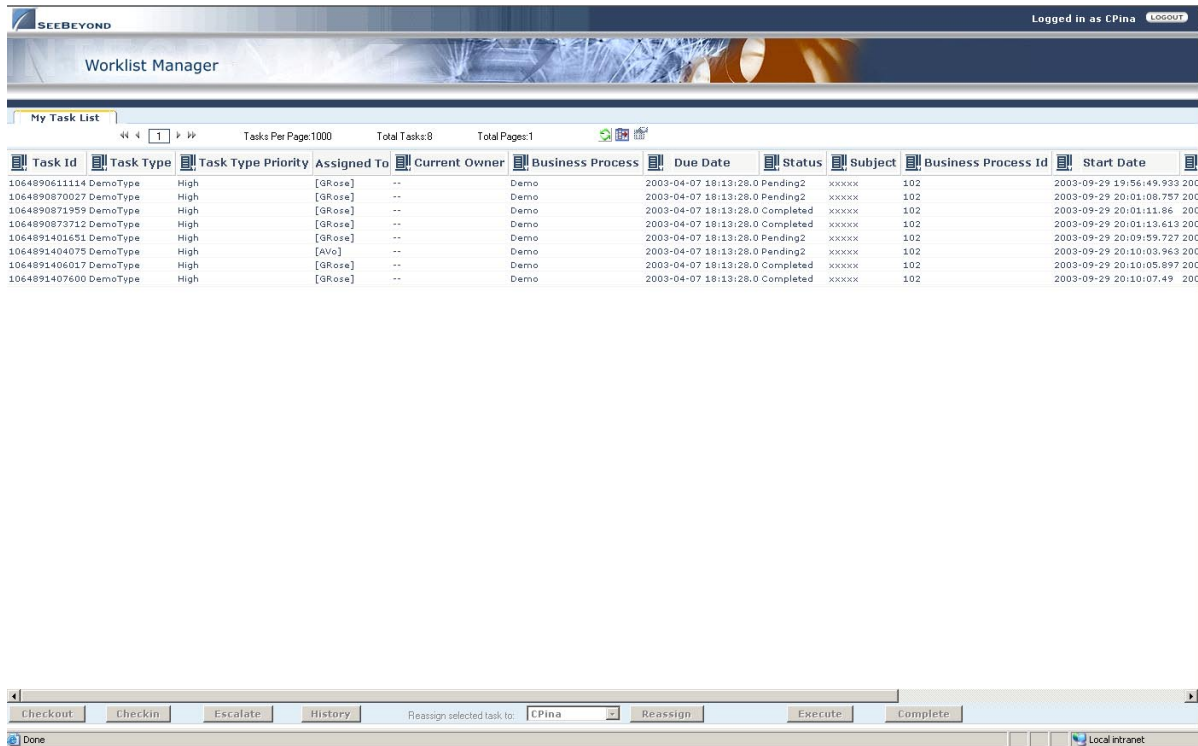
Here is an explanation of the options you will see as a manager:

- **Checkout/Checkin** is necessary to ensure that more than one person is not working on the same task. This is the first thing you must do before you can make any changes to a task.

- **Escalate** sends the task to the user's manager.

- **History** provides a record of all past information about the task.

- **Reassign** allows you to pass a task to another team member. Click **Reassign** and select a name from the drop-down list.

- **Execute** opens the task so that you can perform the task.

- **Complete** commits the changes. You must select **Complete** before leaving the Worklist Manager or your changes will be lost.

4.3 Business Process Properties

Each Business process has a set of properties that you can change and create. These properties provide rapid creation and deletion of business process attributes. eInsight ESB uses this information to automatically create the appropriate business process attributes and input/output structures, for use in the Business Rule Designer.

Edit Business Process Properties

1   Right-click on the **Business Process** you want to edit.

2   Select **Open Business Process Properties Sheet**.

The **Business Process Properties** dialog appears as shown in Figure 13.

Figure 13   Business Process Properties: General Tab

4.3.1. General

The General Tab is the first tab you see when you begin to edit a business process property. You can change the business process name, edit the target namespace and select the Persistence State from this tab.
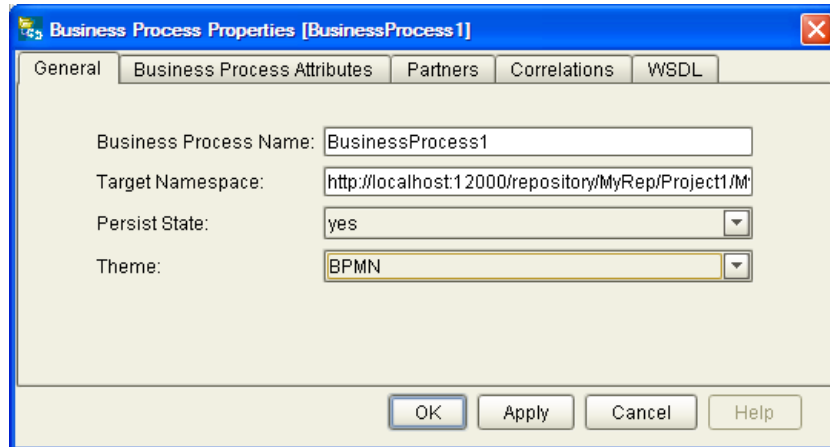
Edit General Business Process Properties

1   From the **General** tab, you can edit:

   ◆ **Business Process Name -** Change the default name.

   ◆ **Target Namespace** - address of the business process.

   ◆ **Persist State** - See **"Persistence" on page 63** for more information.

   ◆ **Theme -** The default Theme is BPMN. Select Custom 1 for a different look.

2   Click **OK** to Save your changes and exit the **Business Process Properties** dialog box.

Business Process Attributes

Business Process Attributes are data values used by a business process. They make it possible to share data between activities in a business process as well as move data to and from the components that implement those activities. Complex structures such as OTDs and Collaborations are represented automatically in the Enterprise Explorer and are available for use in your business process.

Some examples of Business Process Attributes are:

- customer names
- addresses
- order quantities
- item descriptions

Business Process Attributes are used to pass values between the business process and external sources. Business Process Attributes can also be assigned to specific activities. For example, the customer name is passed to an order process from the originating source. The customer name may be used by several of the activities in the business process and is included in the business process output.

eInsight ESB can pass all or part of a complex structure or it can even assemble a composite input to a component or Web Service from multiple business process attributes.

Create New Business Process Attribute

1 Select the **Business Process Attributes** tab (see Figure 15).

2 Select **New** to add a **New Business Process Attribute**.

The **New Business Process Attribute** dialog box appears as in Figure 14.

Figure 14 New Business Process Attribute
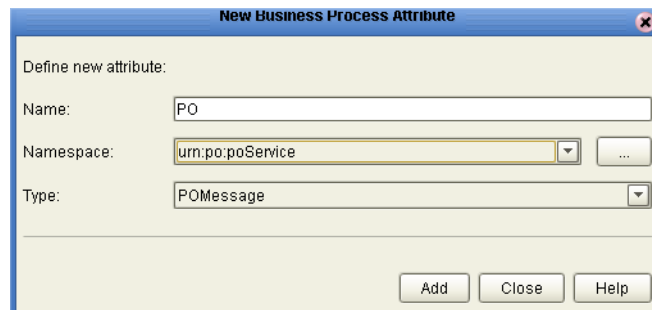
3 From this dialog, complete the following information:

- Enter a **Name** for the attribute.
- Select or browse for an existing **Namespace.**
- Select an available **Type** for your attribute.

4 Click **Add** to Save the attribute or Close to return to the **Business Process Properties** dialog box.

Edit a Business Process Attribute

1 Select the **Business Process Attributes** tab (see Figure 15).

Figure 15 Business Process Properties: Business Process Attributes Tab

2   Select an existing attribute and:

◆ **Rename**: Select and double click the attribute name to rename it.

Note: Some attributes cannot be renamed.

◆ **Delete**: Select Delete to remove the attribute.

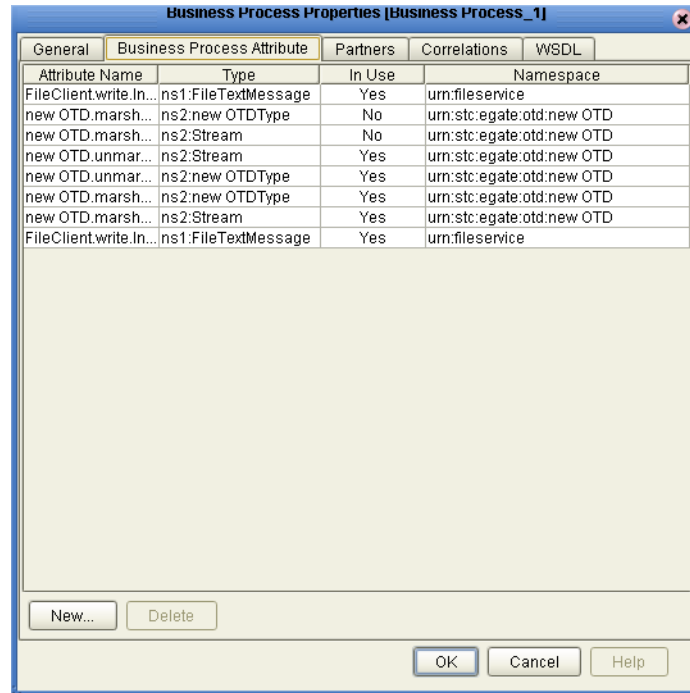3   Click **OK** to Save your changes and exit the **Business Process Properties** dialog box.

4.3.2. Partners

The Partner is an abstracted identification for an external system that will appear in the Binding dialog within the Connectivity Map Editor. Multiple activities can use the same external system – hence, multiple Activities may have the same Partner. By default, eInsight ESB assigns this identification to speed up and automate the model development.

Create New Partner

1   Select the **Partner** tab (see Figure 16).

Figure 16 Business Properties: Partner Tab
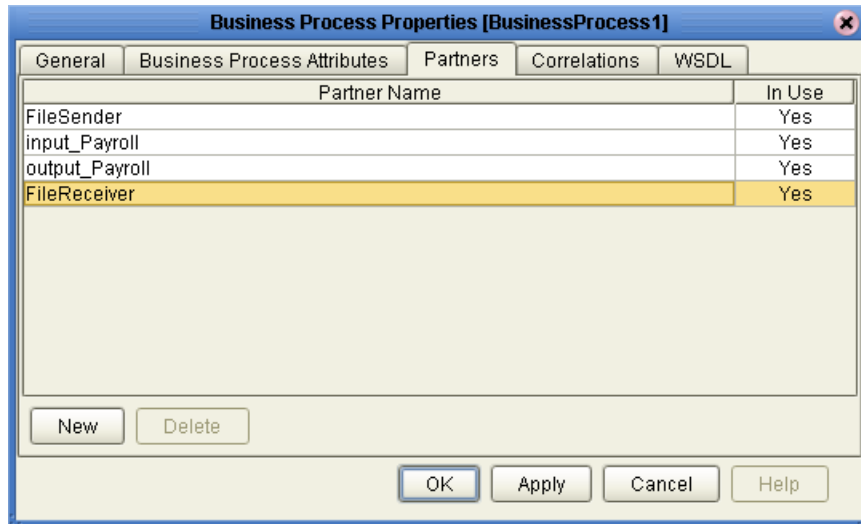
2 Select **New** to add a **New Partner**.

The **New Partner** is added to the Partner list, as shown in Figure 17.

Figure 17 New Partner
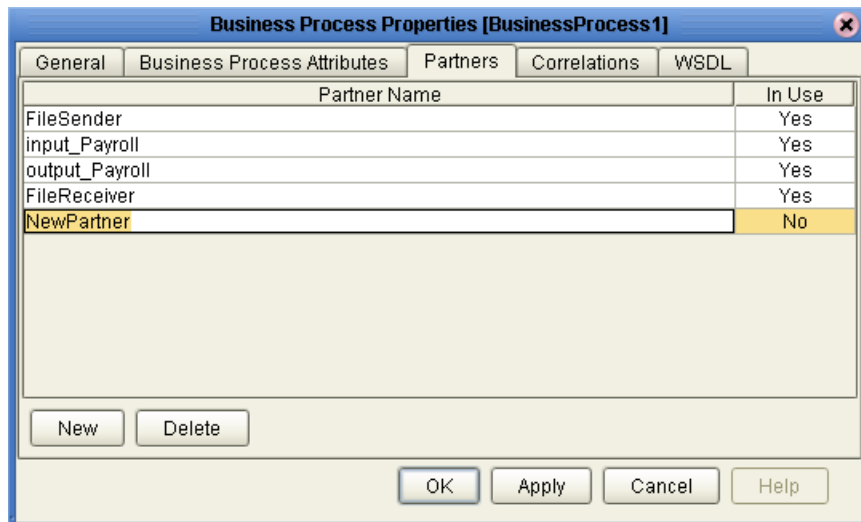
3 Click the Partner name to rename the Partner.

Delete a Partner

You can only delete a Partner that is not in use.

1 Select the **Partner** tab (see Figure 16).

2 Select the Partner name that you want to remove.

3 Select **Delete** to remove the Partner.

The **New Partner** is removed from the Partner list.

Select a Partner for an Activity

1 Select an activity from the Business Process.

2 Click the Show Property Sheet icon from the Business Process Designer toolbar.

The activity's property sheet appears, as shown in Figure 18.

Figure 18   Activity Properties

3 Click the Partner field and select or change the default Partner from the drop-down list.

4.3.3. Message Correlation

In some situations, you want to impose conditions on a set of messages, process a group of messages together or make a decision contingent on the receipt or non-receipt of all messages of a certain type. These situations can be handled using Message Correlation.

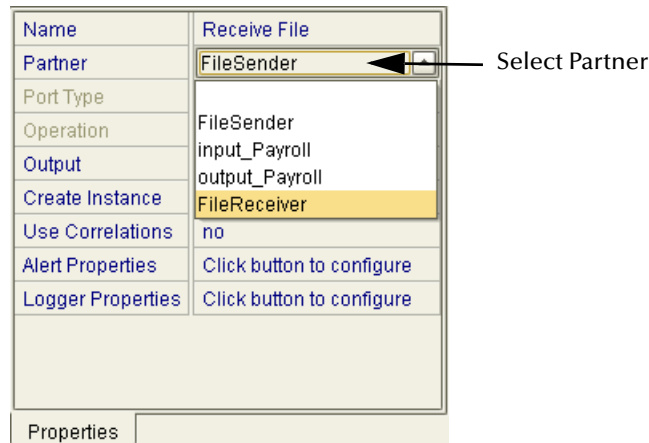You can use Message Correlation when you want to sort messages into separate groups and execute your business rules on a group-by-group basis rather than an message-by-message basis. Messages in the same group are linked.

Groups are stored in memory, to facilitate processing each one as a unit. These groups are called containers. As eInsight ESB retrieves a message, it correlates the received message against a business process instance. If found, eInsight ESB stores the message in the container for that business process. Otherwise, it will instantiate a new instance of that business process.

Message Correlation Example

In this example, the business process, expects to receive three course grades. The courses are Math, English and Computer Science. Each message contains the course grade, the course type, and a Correlation ID to indicate where this message belongs.

A new message arrives with a Correlation ID of 101. The first thing eInsight ESB does is correlate that message to see if there is a match on the newly arrived message. Since this is the very first message, there is no match and a new instance is created. The second

message has a Correlation ID equal to 101 and is forwarded to the same business instance as above. The third message has a Correlation ID of 102. Thus it is forwarded to a new business instance and so on.

This process can continue based on conditions set by the user. This is based on Count or time expiration. A sample business process is shown below for Message Correlation. The first business process uses a File eWay to read a DTD based message, unmarshal it and then invoke the second business process passing in the unmarshaled message.

Figure 19   Message Correlation: First Business Process

The second business process receives the unmarshaled message using Event Based Decision and Timer Events. The Event Based Decision and Timer Events are in a While loop. The While continues to loop until either a count has been reached or time has expired. When a messages are received, they are stored in containers.

Figure 20   Message Correlation: Second Business Process
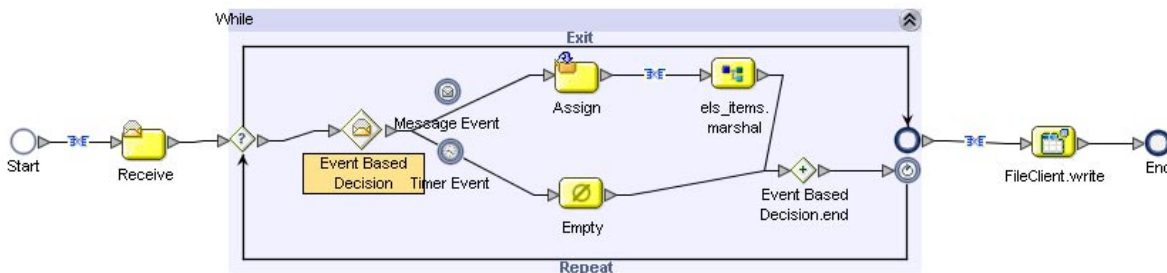
The Timer Event is used to set the expiration time. If time expires, then the loop condition is set to false to terminate the loop. If a message is received, then message counter is incremented and if the maximum number of messages have been received, then loop is terminated. At the end, the date is written to a file.

4.3.4. Correlation Keys and Sets

Configuring Correlation consists of the following steps:

- **"Create a Correlation Key" on page 40**

  A *correlation key* is a value that you can assign to a business process, like a Purchase Order number. The correlation key provides a way to associate and route information about specific business process instances. For asynchronous message

exchange between components, you must implement correlation of the instance identification. An example of when you use asynchronous message exchanges is when you create a Receive activity in the middle of a business process.

▪ **"Add Correlation Sets" on page 41**

Correlation sets are groups of properties shared by all messages in the group. A correlation set matches messages and conversations with a business process instance. For example, you may wish to assign a Purchase Order number and an invoice number to a transaction, so that all information about the purchase and payment are associated.

▪ **"Bind Sets of Correlations to Activities" on page 42**

When using one or more correlation sets within a business process, the values must be initialized at some point. If the user chooses to initialize the set within an Activity, they will also identify which Business Process Attribute will be used (or both).
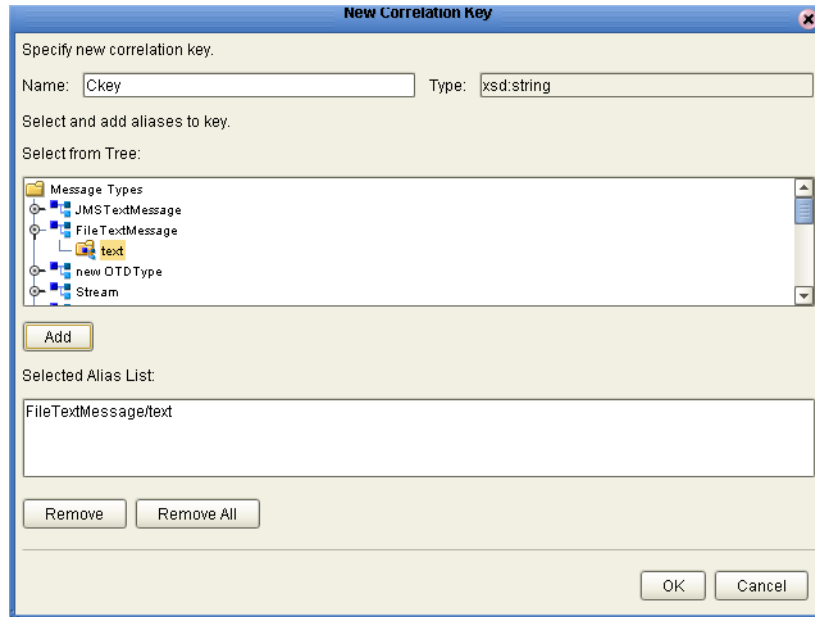
Create a Correlation Key

1   Select the **Correlations** tab (see Figure 21).

Figure 21   Business Process Properties: Correlations Tab
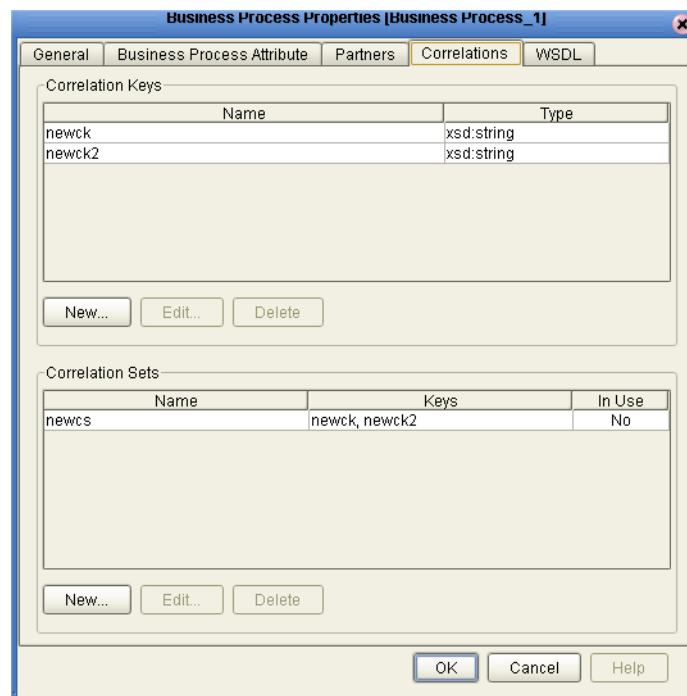
2   Select **New** from the **Correlation Keys** section of the dialog box.

The **New Correlation Key** dialog appears as shown in Figure 22.

Figure 22 New Correlation Key Dialog Box

3 From the **New Correlation Key** dialog box:

  A Enter a **Name** (alias) for the Correlation Key

  B Select a **Message Type** from the list to alias. Select one or more correlation keys that comprise a unique identifier for a step in a business process.

4 Click Add to save the new alias to the **Selected Alias List**

5 Click **OK** to save your changes and exit the **New Correlation Key** dialog box.

Add Correlation Sets

  1 Select the **Correlations** tab (see Figure 21).

  2 Select **New** from the **Correlation Set** section of the dialog box.

   The **New Correlation Set** dialog appears as shown in Figure 23.
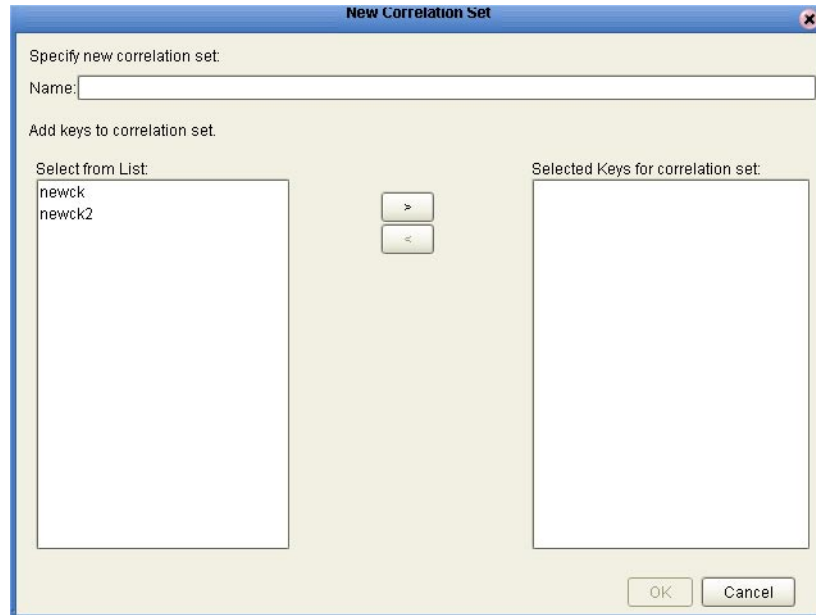
Figure 23 New Correlation Set Dialog Box

3   From the New Correlation Set dialog box:

   A   Enter a **Name** for the new Correlation Set.

   B   Select **Correlation Keys** from the list to add to the Correlation Set.

   C   Click the arrow button to move your selections to the Correlation Set.

4   Click **OK** to save your changes and exit.

Bind Sets of Correlations to Activities

1   Select an Activity.

2   Select **Show Property Sheet** from the toolbar.

3   Locate **Use Correlations** and select **Yes.**

4.3.5. **WSDL Files**

WSDL files are used to invoke and operate Web services on the Internet and to access and invoke remote applications and databases.

The WSDL tab is available from the Business Process Properties dialog. From this dialog, you can upload a WSDL file with predefined Business Process Attributes for use in your business process.

Upload a WSDL File

1   Select the **WSDL** tab (see Figure 24).

Figure 24   Business Process Properties: WSDL Tab
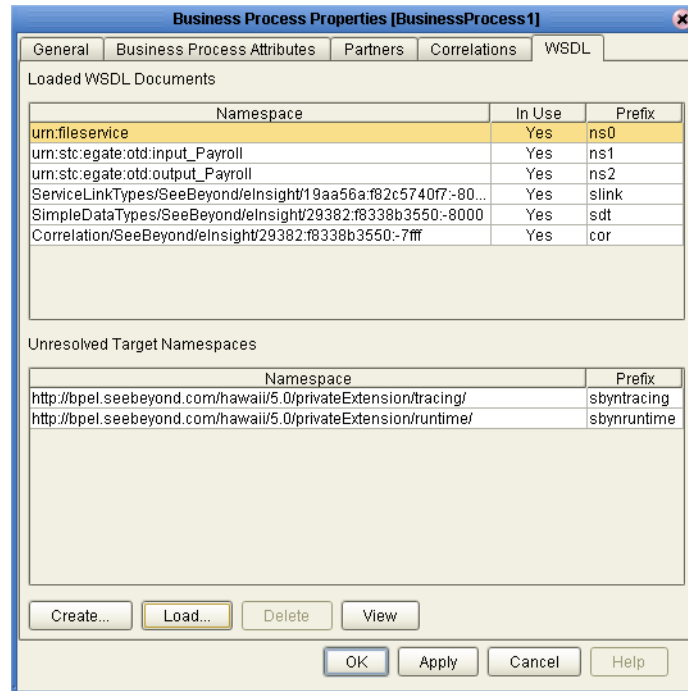
2 Click **Load** to upload a WSDL file.

The **Load WSDL** dialog box appears, as shown in Figure 25.

**Figure 25**   Load WSDL dialog box
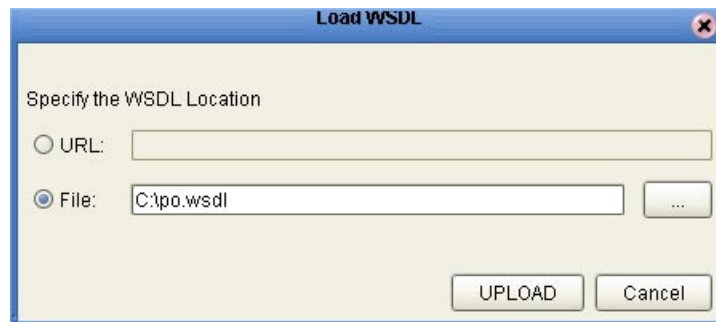
3 From the **Load WSDL** dialog box:

A Select **URL** or **File**, to choose where your WSDL file is located.

B Enter the path to the WSDL file.

You can also use the browse button to locate a File location.

4 Click **UPLOAD**.

WSDL Interface Designer and Viewer

Create a New WSDL File

1 Click **Create** to create a new WSDL file.

2 The **WSDL Interface Designer** appears, as shown in Figure 26.

Figure 26 WSDL Interface Designer
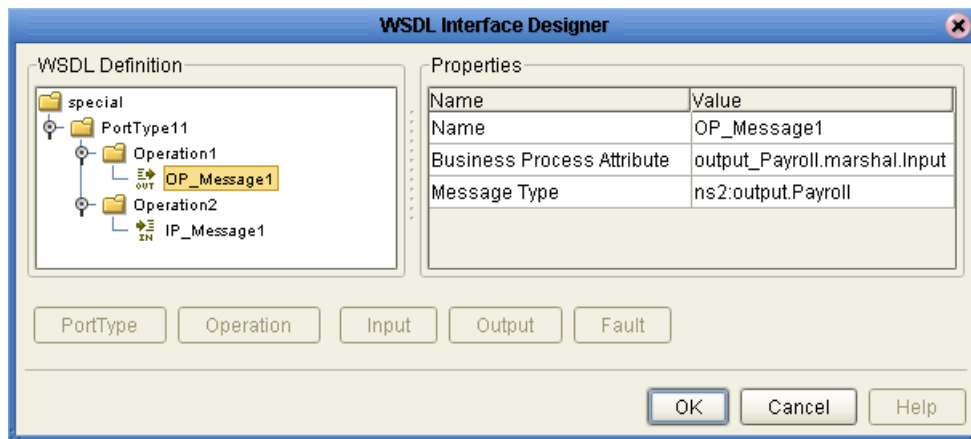
3 Specify the following:

- PortType

- Operation

- Input and/or Output and/or Fault

4 Click on each WSDL definition to select the Business Process Attribute and Message Type that you will use.

Edit a WSDL file

1 Select the WSDL tab from the Business Process Property sheet.

2 Select a WSDL file from the list and click Edit.

3 The WSDL Interface Designer appears, as shown in Figure 26.

View a WSDL file

1 Select the WSDL tab from the Business Process Property sheet.

2 Select a WSDL file from the list and click **View**.

3 The WSDL Viewer appears, as shown in Figure 27.

Figure 27 WSDL Viewer

4 From the WSDL Viewer you can copy all or part of the WSDL code to the clipboard, and paste it to a text file.

You cannot edit from the WSDL Viewer.

4.3.6. Sub-Processes

Sub-processes are deployed business processes that can be invoked within the same Integration Server. When a sub-process is dropped into a business process, the Business Rules Designer is used to configure the input and output attributes for the sub-process.

Sub-Process Overview

- Create a Business Process (this is the parent process). Create an Empty Activity as a placeholder for the sub-process.

- Create a second Business Process (this is the sub-process). Create an empty Receive Activity at the beginning of the Business Process and Reply Activity at the end.

- Create or upload a WSDL file that represents the inputs and outputs of the sub-process.

- Create a Partner from the Business Process Property Sheet: Partner tab.

- In the sub-process, open the properties of the initial Receive and select the appropriate Partner, portType, and operation. Do the same for the Reply.

Note: *If the sub-process is to be invoked as a synchronous request/reply web service, the Receive and Reply must have the same partner, portType, and operation.*

- Generate the Business Process Code and save the sub-process.

- Return to the parent process from the Project Explorer, drag the first operation under the sub-process to the sub-process placeholder in the parent business process.

Note: *It is not always the first operation - it is the operation assigned to the Receive/Reply in the sub-process.*

- Perform any input/output mappings using the Business Rule Designer, in the parent business process.

4.3.7. Repeating Nodes

For Web Services/components that contain repeating nodes, the Business Rule Designer displays repeating nodes within the input/output Attributes for each Activity. Repeating nodes contain the repeating icon. For direct node mapping, repeating nodes are used to dynamically populate 1-n values based on the runtime data.

Mapping Elements within Repeating Nodes

1 From the Method Palette, in the Business Rule Designer, select the **copy-from** method and drop it into the operator canvas.

This action opens the **copy-from** entry box.

2 Enter the XPath expression and link it to the input of the appropriate activity.

The link from the appropriate output is implied within the XPath expression.

XPath Predicates

The XPath predicate functionality allows the user to isolate particular elements within repeating nodes at runtime.

When assigning values in the Business Rule Designer, you can access a *predicate* operator to extract particular elements from repeating nodes within a business process attribute. The Attribute uses the repeating characteristic of the node from the underlying OTD. The operator includes an example for the xpath syntax. At runtime, the XPath expression selects the correct element and performs the mapping as designed.

4.4 Exception Handling

This section explains the concept of exception handling and how to configure various methods of handling errors.

- **"Scope" on page 47**
- **"Compensation" on page 49**

4.4.1. Overview

Exception handling is the identification of failed components or systems. In eInsight ESB, Exception handling allows one or more components to throw an exception that is caught by eInsight ESB within a *scope*. Using the scope element, you can configure eInsight ESB to catch all exceptions or certain exceptions that you specify. The elements that you use to configure Exception handling in your model are:

- **Catch Named Exceptions**
- **Catch All Exceptions**

Exception handling in business processes relies heavily on the concept of *compensation*. Compensation is an application-specific activity that reverse the effects of a previous activity that was carried out as part of a larger unit of work that is being abandoned.

Business processes are often of long duration and use asynchronous messages for communication. They also manipulate sensitive business data in back-end databases and line-of-business applications. As a result, the overall business transaction may fail or be cancelled after many transactions have been committed during its progress. In these cases, the partial work may need to be reversed.

4.4.2. Exception Handling Configuration

Exception handlers are configured to catch errors that are thrown by eGate components and/or Web Services. These systems can be configured to publish one or more exceptions.

- **Manual Exception Handling**: The model can contain business process logic designed to handle the exception.
- **Automatic Exception Handling**: Pre-packaged functionality guides the user to create multiple types of catches for thrown exceptions.

Each exception can be handled differently. This is one example:

1 Build the exception handling logic as a business process.

2 Select the exception handler to configure which exception triggers the exception handling process.

3 Drag the Scope element to the Business Process Designer.

4 Drag the Exception modeling element into the scope for which it should take effect.

5 Define a business process that appropriately handles each exception.

6 Configure the exception handler to take place when one of the components within the Scope throws the appropriate exception.

4.4.3. Scope

Scope allows you to define a range

- For handling of exceptions
- For creating compensation logic

The range of the Scope can span one or more Activities in the business process or even the entire business process.

Scope and Process-Level Exceptions

Either the Named or Catch All exceptions can be used at the business process level.

Named exception

1 Drag the **Catch Named Exception** element into the Scope for which the exception handler applies.

2 In the Exception Handler properties, configure the following:

- ◆ The **Fault Container** – which is the output Attribute that will be containing the runtime name of the thrown fault.

- ◆ The **Fault Name** – which is the runtime value for the exception that will be passed from the component at runtime.

Note: *The fault name is auto-populated with values based on the components dragged to the Business Process Designer.*

Catch All Exceptions

To use the Catch All Exceptions element:

- ▪ Drag the Catch All Exceptions activity to the a scope in the Business Process Designer or just onto the canvas. This will capture any and all exceptions that occur.

- ▪ Configure the Exception Handler to perform an action when an exception is encountered.

Figure 28   Catch All Exceptions
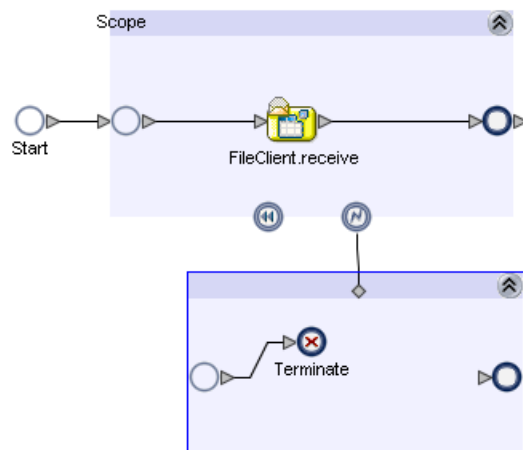
4.4.4. **Compensation**

Compensation allows the modeler to create the process flow for executing complex compensations. Exception Handlers for parent scopes invoke the correct Compensation Handlers in the appropriate order.

Using Scope and Exceptions to Trigger Compensation

- Compensation Activity – used in an exception handler to initiate the compensation process. The Compensation Activity indicates where the Compensation should be initiated.

- Model the Compensation process as a business process.

- Compensation Handler – drop within a scope to create the compensation logic for a given scope.

<div style="text-align:right">

Chapter 5

</div>

Deploy and Monitor Business Process Models

5.1   Overview

Topics in this chapter are:

- **The Business Process and the Connectivity Map** on page 50
- **Deploy a Business Process** on page 52
- **Persistence and Monitoring** on page 52

5.1.1. The Business Process and the Connectivity Map

The Connectivity Map represents connection information in the ICAN Suite. The flow is represented at a higher level than in the Business Process Model. eInsight ESB also uses the information in the Connectivity Map to establish and maintain connections to systems for the correct step in a business process.

Creating the Connectivity Map that Includes a Business Process

1 Drag the desired Business Process from the Enterprise Explorer to the Connectivity Map Editor.

2 Add the external systems and components to the Connectivity Map Editor, as shown in Figure 29.

Figure 29   Connectivity Map with Business Process
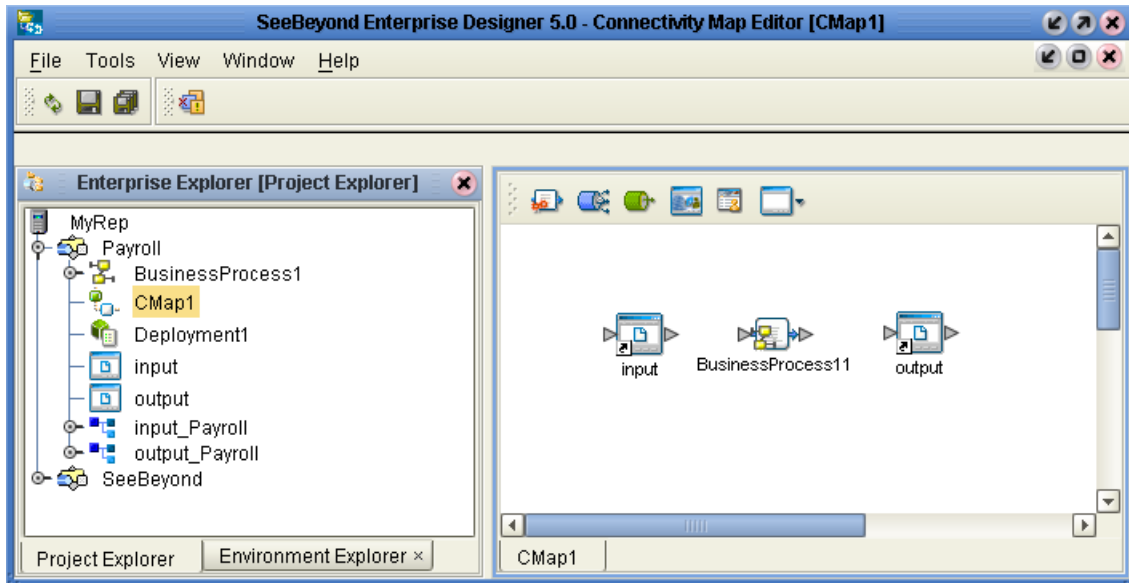
Open the Business Process to Connect the Activities

1   Double-click on the Business Process in the Connectivity Map to open the Binding Dialog.

2   Connect the Activities to the appropriate component, as shown in Figure 30.

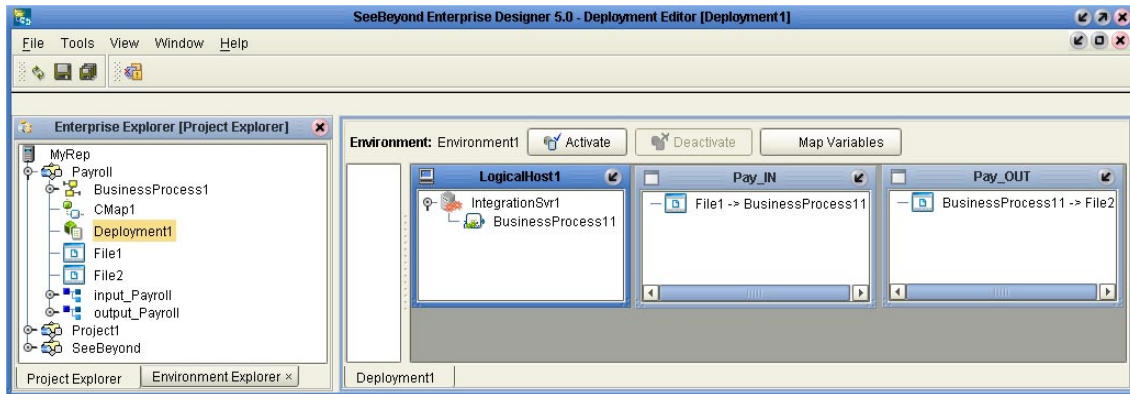**Figure 30**   Connectivity Map: Business Process Binding
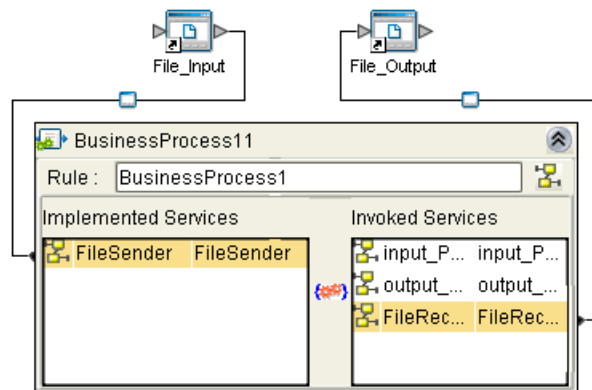
   ◆ Receive Activities appear in the left pane.

   ◆ Invoke and Reply Activities appear in the right pane.

5.1.2. **Deploy a Business Process**

A business process is like any other ICAN Suite component. After creating your Environment, Logical Host and other necessary hosts, create a Deployment Profile.

Figure 31   Deployment Profile

The deployable components along with the Business Process from the Connectivity Map will appear in your Deployment Profile. Drag and drop the Business Process to the desired Integration Server. Activate your Deployment Profile to complete the deployment of the components to the target hosts. For more information about Deploying a Project, see the *eGate Integrator User's Guide*.

5.2   **Persistence and Monitoring**

This section discusses Persistence and Monitoring, which both require a database. eInsight ESB contains database scripts to create the database schema. The database schema allows you to collect and persist data from your business process. Once the data is persisted, you can also use the Enterprise Manager to monitor business processes.

Topics in this chapter are:

- **"Persistence" on page 52**
- **"Monitoring" on page 56**

5.2.1. **Persistence**

The main areas to be configured in order to have persistence are:

- **Runtime Configuration** on page 53
- **Download Database Scripts** on page 55
- **Run the Database Install Script** on page 56

Runtime Configuration

To configure the eInsight ESB database

1 Right-click the **Integration Server** (IS) in your Environment from the Environment Explorer and open the tree structure that contains properties.

2 Navigate to, and select the **eInsight.**

The properties dialog appears for eInsight, as shown in .

Figure 32   eInsight Configuration
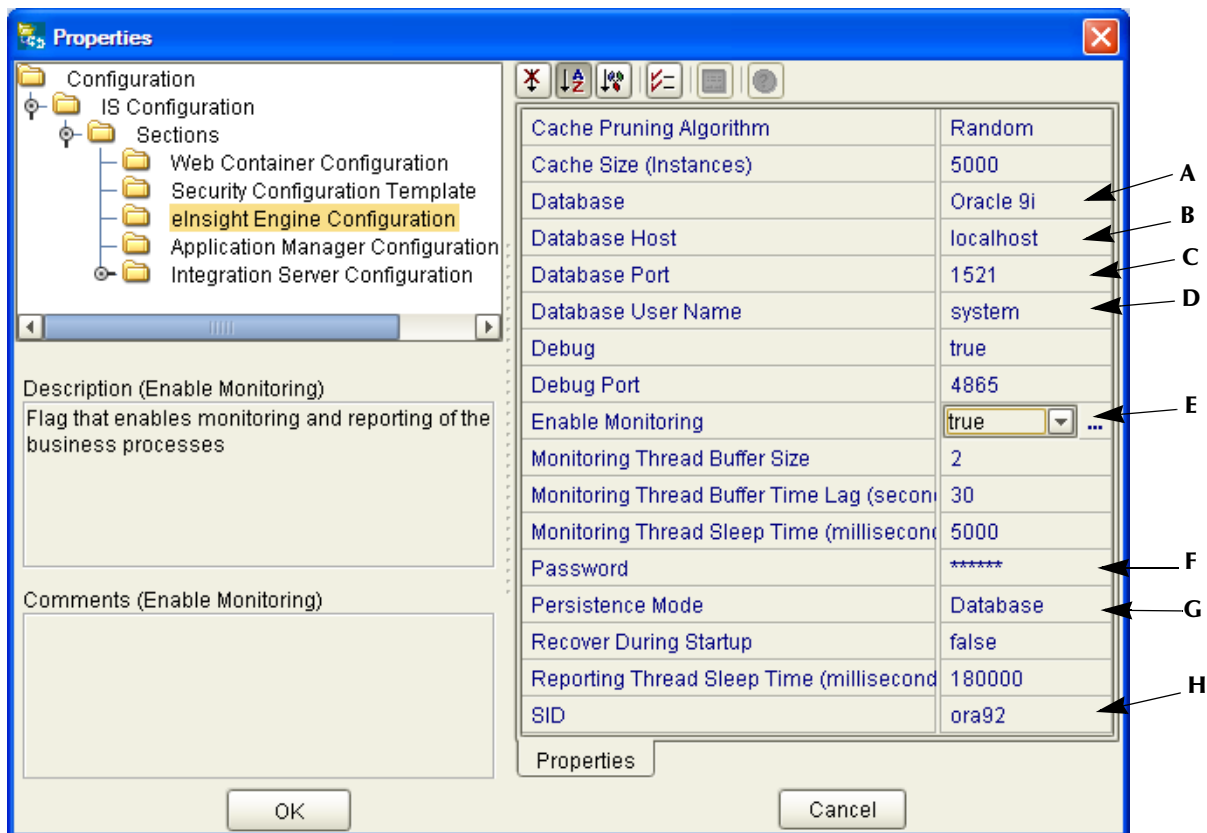
3 Configure the following to connect to a database:

A **Database** - Select Sybase 12.5, Oracle 8.1.7, Oracle 9i, or SQL Server 2000.

B **Database Host** - Enter the name of the machine where your database resides.

C **Database Port** - Enter the port number to connect to your database.

D **Database User Name** - Enter the User Name for your database.

E **Enable Monitoring** - Select True to use Monitoring.

F **Password** - Enter the password for your database user.

G **Persistence Mode** - Select Database to use persistence.

H **SID** - Enter the database name or SID.

Note: *The other eInsight settings can be left as default, or you can change the options to your preference.*

5.2.2. Configure Persistence for the Business Process

1 Right-click your Business Process and select **Open Property Sheet.**

2 Select **Yes** for the **Persist State** option, as shown in Figure 33.

You set persistence state for individual business processes. The default setting is no.

**Figure 33** Business Property Sheet
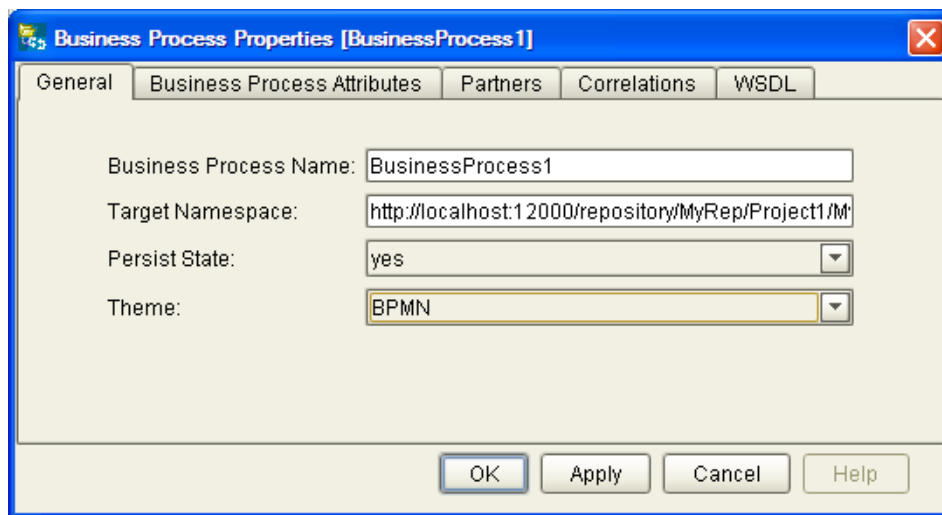
3 Click the **Save All** toolbar icon.

This creates a **Database Install Script** option under your Business Process.

Configure connection information

1 Right-click the **Database Install Script** under the Business Process and select **Properties**.

2 Fill in the connection information for your database (as shown in **Figure 34 on page 54**).

Figure 34 Properties of Database Scripts
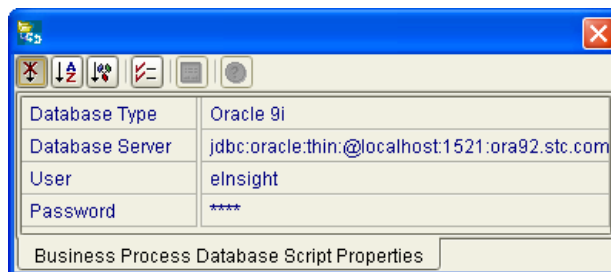
Run the Database Install Script

1   From the Project Explorer, expand your Business Process.

2   Expand the **Database Install Scripts** folder.

3   From the **Database Install Scripts** folder, right-click the appropriate database and select **Run.**

The scripts complete the database creation process.

Uninstall the Database Script

1   From the Project Explorer, expand your Business Process.

2   Expand the **Database Scripts** folder.

3   From the **Database Scripts** folder, right-click the appropriate database and select the **Uninstall** option**.**

5.2.3. Download Database Scripts

This is an alternative method to **"Run the Database Install Script" on page 55**. You may wish to download the database scripts and run them outside of the ICAN environment.

Download the compressed script files

To create the runtime recoverability database schema, you can download a Database Script that is automatically installed with eInsight.

1   Expand the **SeeBeyond** folder in the Project Explorer.

2   Expand the **eInsight** folder that is located under the SeeBeyond folder.

3   Expand the **DatabaseScripts** folder.

4   Select the appropriate database and expand the corresponding folder (Oracle, SQLServer or Sybase).

5   Right-click the .zip file associated with the appropriate database (Oracle.zip, SQLServer.zip or Sybase.zip) and select **Export**.

6   Unzip the database script to a local folder.

Included in the zipped file are:

- **install_db.bat** - This script will create the tablespace, users, tables, stored procedures, and any initial value.

- **uninstall_db.bat** - This script reverses what the install_db.bat script creates (drops tables and users, deletes stored procedures).

- **database specific sql scripts** - These scripts are called by the install_db.bat and uninstall_db.bat commands (such as, create_tables.sql, drop_tables.sql, etc.)

- A **Readme.txt** file with additional instructions, specific to your database application.

Run the Database Install Script

If you are using a Sybase or SQL Server database, you need to modify the script before you run it.

Modify Sybase or SQL Server Script

1   Open and edit the install_db.bat.

2   Enter the database user and password that has privileges to your Sybase or SQL Server database.

To execute install_db.bat

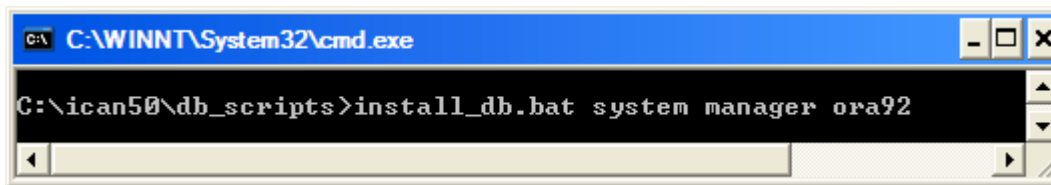1   Open a command window and navigate to the directory where **install_db.bat** is located.

Important: *The database user that executes these scripts must have permission to create tables.*

2   Enter the following at the command prompt, as shown in Figure 35:

```
install_db <user> <password> <tns>
```

  ◆ <user> is the database username

  ◆ <password> is the database user password

  ◆ <tns> is the database or tns name

Figure 35   Install_db.bat

The default user and password created from these scripts is "einsight". You can modify the user, password, disk space allocated for tables, and user permissions. The table and column definitions should not be modified.

5.2.4. Monitoring

Once persistence is configured, you can use the Enterprise Manager to monitor your business process instances.

To monitor your business process

1   Create a number of instances in your business process.

2   Launch the **Enterprise Manager**.

3   Select the **Home** tab.

4   Select the **Monitor** icon to bring up the tree structure which allows you to navigate through projects or environments.

5   Select the **Projects** tab.

6   Navigate to the correct **Project/Deployment Profile/Connectivity Map**, and select the **Business Process** name.

Figure 36   Monitor View
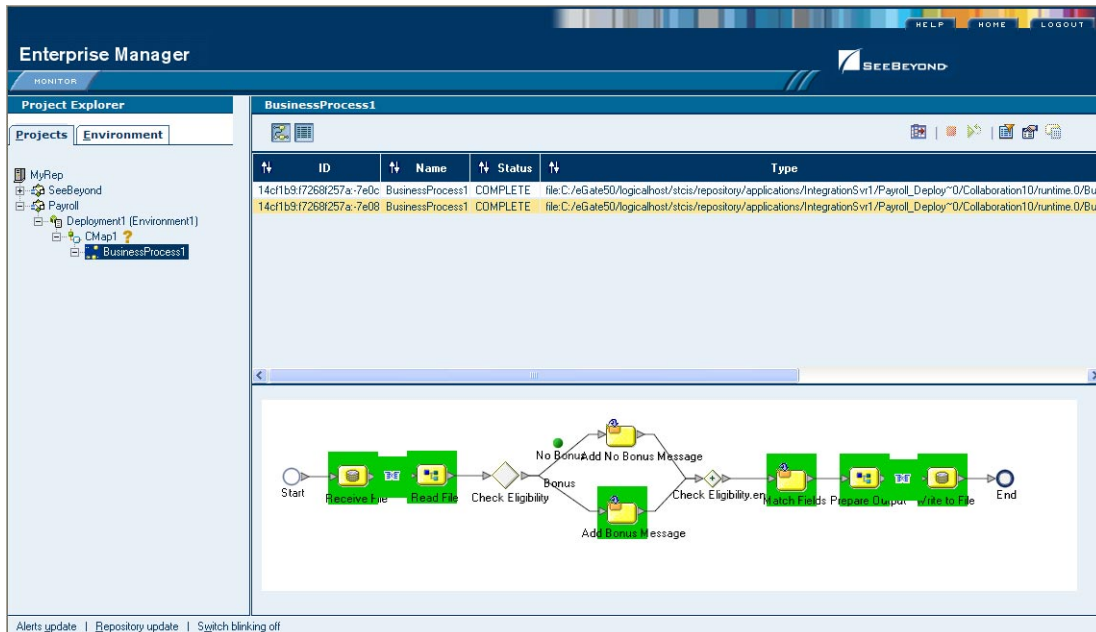
Monitor options

In the right pane of the Enterprise Manager, you can see the model and/or the instance list – the toggle at the top of the pane controls the view.

The options are:

- show the model
- show the instance list
- show the instance list and model

Exception Management

Exception Management allows users to quickly identify and rectify problems with components or systems. From the Enterprise Manager, you can double-click on business process components to go directly to the problem.

From the Enterprise Manager (Monitoring interface), you can:

- Filter the list of displayed instances to quickly identify exceptions.
- Navigate to particular versions of a Business Process to monitor the progress of instances.
- Use a Web based interface allows users to securely access the monitoring environment over the internet.
- Identify exceptions

Failed components/systems create visual alerts through the business process monitoring interface. The integrated monitoring environment allows you to determine the problem, assign a resource to fix the problem, and if necessary, restart the affected instances. You can quickly identify troubled instances from a large number of instances, repair and restart that instance for continued processing.

<div align="right">

Chapter 6

</div>

Implementation

The case study in this chapter is designed to illustrate the functionality of eInsight, in addition to showing a working example of a business process implementation.

6.1 Overview

Implementing an eInsight ESB system is the process of translating the vision of the business user into a functioning system. You implement an business process model by using the modeling components. Modelling components are mostly pre-configured but some may require modification.

6.1.1 Road Map

Each type of implementation involves a different approach, however, there are certain similarities. To give you an overview of the complete process, the following implementation road map contains high-level steps for this implementation. The road map is further refined and given more detail in the case study that immediately follows.

Figure 37, illustrates the major steps in the integration process for this implementation.

Figure 37   Integration Road Map
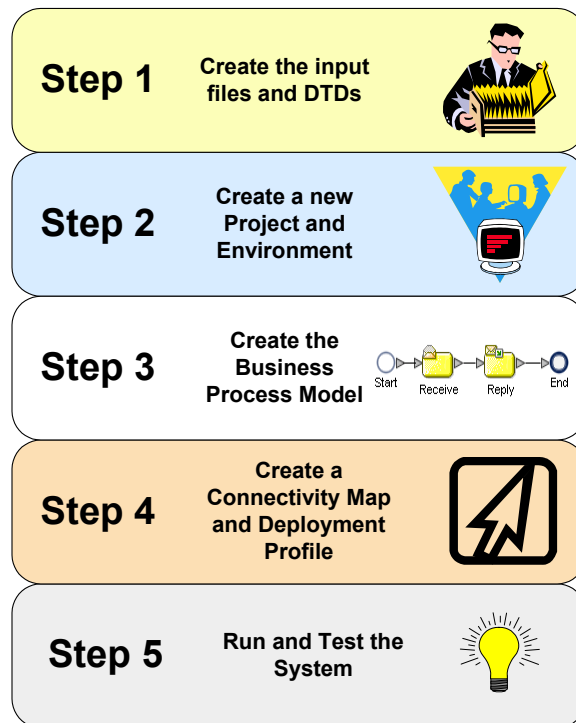
Step 1: Create the Input Files and DTDs

The first step in this implementation requires that you create two input files for the system. In an actual implementation, your input files may come from an external system.

The first step also entails creating your Document Type Definition (DTD) files. The DTD file tells the system which elements it should expect from the input files and how to format the output data.

Step 2: Create a New Project and Environment

In the second step of this implementation, you will create a new Project where your business process will reside and a new Environment for your Project.

Step 3: Create the Business Process Model

In this step, you will create a new business process, add the modelling elements and link them together. You will also configure the modelling elements and links to process the data.

Step 4: Create a Connectivity Map and Deployment Profile

When you create the Connectivity Map, you are making the connections between the system components and the external systems. You will also create a Deployment Profile that you will activate when you put the system into production.

Step 5: Run and Test the System

To run the system, you must invoke the Bootstrap. This action will pick up your Deployment Profile and execute your business process. Once the system processes your input files, an output file is created. To verify that this implementation has completed properly, check the output file.

6.2 Case Study: Payroll Processing

This case study begins with a description of the scenario and then shows how to set it up. The case study discussed in this chapter illustrates a simplified implementation of payroll processing. In this case, eInsight ESB receives payroll data as XML files.

Once eInsight ESB has received the data, a check is made to see if the employee is eligible for a bonus, if they are, the bonus is set. Finally, the payroll is processed and a message added to the paystub, indicating whether a bonus was paid. Figure 38 shows the components involved in the business process implementation.

Figure 38 Data Flow
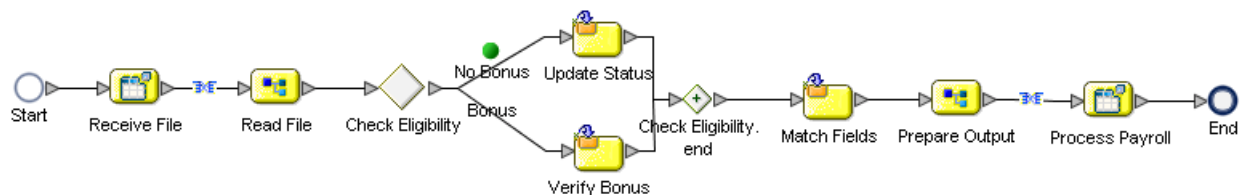
1 The first File eWay picks up the input XML files containing the employee's information from a local folder on your computer. The payroll information is used to start a business process instance. The engine retrieves the information and uses it to execute the decision logic.

2 The engine uses the decision logic information it contains to check the employee's probation status and continues along one path or the other, depending on that status.The decision logic determines whether the employee is eligible for a bonus, and then moves forward to the next activity in the business process based on the result.

3 If the employee is eligible for a bonus, the next activity is **Set Bonus**; if the employee is not eligible, the next activity is **Update Status**.

▪ Let's assume the employee is ineligible for a bonus because they have been employed for less than three months. The Probation status is *true*, therefore the engine proceeds to the corresponding activity, **Update Status,** in the business

process. Once the Comment and Bonus fields are updated, the engine moves forward to the next activity in the business process—**Match Fields**.

▪ Otherwise, the employee is eligible for a bonus and the engine uses the information to verify eligibility. When the **Verify Bonus** activity is finished, the engine moves forward to the next activity in the business process—**Match Fields**.

▪ The **Match Fields** activity uses the Business Rule function to match the data fields in your input file to the data format of your output file.

4 The engine then proceeds to the **Prepare Output** activity and finally the **Process Payroll** activity. **Process Payroll** is a File eWay that performs two functions: it sends a status report to the payroll system, and also writes the data to the output file.

5 The engine has performed the final activity in the business process and completes successfully.

6.2.1 Before You Begin

To complete this exercise, you need to have the following:

▪ ICAN 5.0 products installed:

 ◆ eGate

 ◆ eInsight

 ◆ File eWay

▪ A directory on your local drive named **data**.

Input Files and DTDs

The sample system you are creating requires input information. For this exercise, you will create two input files: **Bonus.xml** and **No Bonus.xml**. These files are in an XML format. You do not need to have an XML editor to create these files. Any simple text editor will work.

The system you are creating also needs a structure for receiving information. That structure is described in the Document Type Definition (DTD) files.

Input XML Files

The XML files that you create here, contain the data that the system receives and changes to create your final output.

Create the Input files

1 Copy the following code sections each to separate text files:

▪ **Bonus.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
  <Payroll>
     <FirstName>Vanessa</FirstName>
     <LastName>Smith</LastName>
     <Probation>false</Probation>
     <Comments></Comments>
```

```
        <Bonus></Bonus>
      </Payroll>
```

▪ **No Bonus.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
   <Payroll>
      <FirstName>Tonya</FirstName>
      <LastName>Lee</LastName>
      <Probation>true</Probation>
      <Comments></Comments>
      <Bonus></Bonus>
   </Payroll>
```

2 Rename the files to **Bonus.xml** and **No Bonus.xml**.

3 Save the files to **C:\data**.

Input and Output DTD Files

Create the Document Type Definitions

These files are later used to define the way data is mapped in the system.

1 Copy the following code sections each to separate text files:

▪ **Input.dtd**

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Payroll (FirstName, LastName, Probation, Comments, Bonus)>
<!ELEMENT FirstName (#PCDATA)>
<!ELEMENT LastName (#PCDATA)>
<!ELEMENT Probation (#PCDATA)>
<!ELEMENT Comments (#PCDATA)>
<!ELEMENT Bonus (#PCDATA)>
```

▪ **Output.dtd**

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT Payroll (FName, LName, Message, BonusTotal)>
<!ELEMENT FName (#PCDATA)>
<!ELEMENT LName (#PCDATA)>
<!ELEMENT Message (#PCDATA)>
<!ELEMENT BonusTotal (#PCDATA)>
```

2 Rename the files to **Input.dtd** and **Output.dtd**.

3 Save the files to a local folder.

6.2.2 Create a New Project and Environment

Create a new Project

1 Launch the Enterprise Designer.

2 Right-click your Repository and select **Project.**

A new Project appears in your Project Explorer tree structure.

3 Rename the Project to **Payroll**.

4 Click the **Save All** toolbar button to save your changes.

Add the Input and Output DTD files to your Project

1 Right-click your Payroll Project and select **New: Object Type Definition**.

The **New Object Type Definition Wizard** appears.

2 Select the **DTD** option from the choices given.

3 Click **Next** to continue.

4 Navigate to the local folder where your **Input.dtd** and **Output.dtd** files are located.

5 Select the **Input** and **Output** DTD files. (You can select multiple files using the CTRL key).

6 Click **Next** to continue.

The **Select Document Elements** dialog appears.

7 Select both **DTD** files.

8 Click **Next** to continue.

The **Select OTD** dialog appears.

9 Click **Finish** (do not change any of the default settings).

Environment

Create a new Environment

1 Select the **Environment Explorer** tab from the Enterprise Designer.

2 Right-click on your **Repository** and select **New Environment**.

3 Right-click on your **Environment** and select **New Logical Host**.

4 Right-click on your **Environment** and select **New File External System**.

The system prompts you to name the **File External System**.

5 Enter **Pay_IN** as the name of your **File External System**.

6 Select **Inbound File eWay** as the **External System Type**.

7 Repeat step 4 and name the **File External System**: **Pay_OUT**.

8 Select **Outbound File eWay** as the **External System Type**.

9 Right-click the **Logical Host** and select **New SeeBeyond Integration Server**.

Your new Environment will look like Figure 39.

Figure 39   New Environment
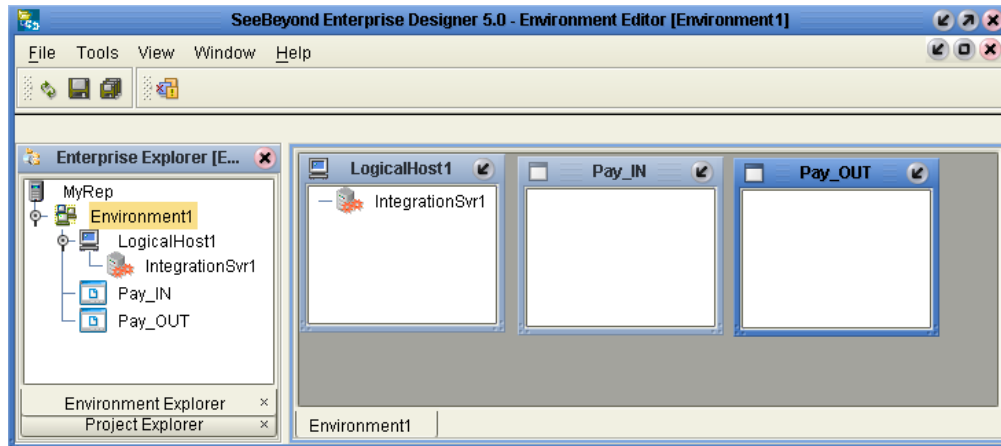
6.2.3 Create the Business Process Model

Create a new Business Process

1   Click the **Project Explorer** tab and right-click on your **Payroll Project**.

2   Select **New**: **Business Process**.

A new Business Process appears in your directory tree under your Payroll Project and a blank Business Process appears in the Business Process Designer (right pane).

Add Modelling Elements to the Business Process Model

This section contains detailed instructions to build your model.

Add the File Receive Activity

This activity uses an inbound File eWay.

1   Double-click the **SeeBeyond Project** from the Project Explorer tree view.

2   Double-click **eWays** under the SeeBeyond Project.

3   Double-click **File** under eWays.

4   Double-click **FileClient** under File.

5   Select and drag the **Receive** activity from **FileClient** to the Business Process Designer.

6   Place the **Receive** activity to the right of the **Start** activity.

Add the Unmarshal Activity

This activity unmarshalls the input data.

1   Double-click your **Payroll Project** from the Project Explorer tree view.

2   Expand the **Input DTD**, located under your Project.

3   Select and drag the **Input Unmarshal** activity.

4 Place the activity to the right of the **FileClient Receive** activity.

Add the Decision Element

The decision element contains logic that determines what will happen to the incoming data. You will configure the logic in a later step.

1 Select **Branching Activities** from the Business Process Toolbar.

2 Select **Decision** from the **Branching Activities** drop-down list.

3 Drag the **Decision** to the Business Process Designer.

The **Decision** element and **Decision.end** appear on the Business Process Designer canvas.

4 Place the **Decision** to the right of the **input _Payroll.unmarshal Activity**. Leave a space between the **Decision** and **Decision.end**.

Add the Business Rule Activities

The Business Rule Activities allow you to map and transform data. You will configure the Business Rule Activities later in this exercise.

1 Select the **Business Rule Activity** from the Business Process Toolbar and drag the activity to the Business Process Designer.

2 Repeat step 1 until you have three **Business Rule Activities** on your canvas.

3 Group the **Business Rule Activities** as shown in Figure 40.

Figure 40 Building the Model
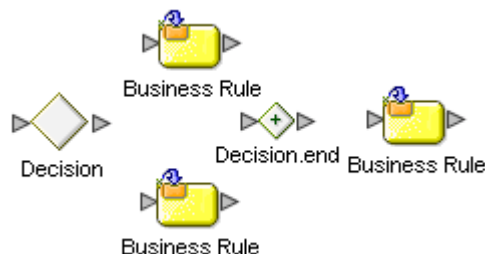
Add the Marshal Activity

This activity marshals the data and prepares it for output.

1 Double-click your **Payroll Project** from the Project Explorer tree view.

2 Click the icon next to **Output Payroll DTD** to expand.

3 Select and drag the Output Payroll DTD **Marshal** operation to the Business Process Designer.

4 Place the activity to the right of the last **Business Rule Activity**.

Add the File Write Activity

This activity is an outbound File eWay.

1 Double-click the **SeeBeyond Project** from the Project Explorer tree view.

2 Double-click **eWays** under the SeeBeyond Project.

3 Double-click **File** under eWays.

4 Double-click **FileClient** under File.

5 Select and drag the **Write** activity from **FileClient** to the Business Process Designer.

6 Place the **Write** activity to the left of the **End** activity.

6.2.4 Configure Modeling Elements

Draw links to connect the model

1 Move your mouse over the **Start** activity until a hand appears.

2 Click and hold your mouse to drag a **Link** between the elements.

3 Repeat steps 1 and 2 to connect the entire model as shown in Figure 41

Figure 41 Linked Model
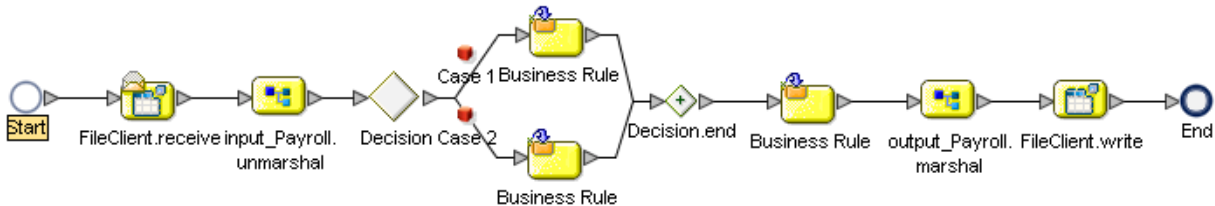
Rename Modeling Elements

You should rename the elements to represent the activity's role in the business process. This makes it easier to understand the model.

1 From the Business Process Designer, click an element label (the name or title under the activity).

A sunken box appears around the label.

2 Type to rename the element.

3 See Table 7 and rename each of the elements as described in steps 1 and 2.

Table 7 Rename Elements

Current Name	Rename to
FileClient.Receive	Receive File
input_Payroll.unmarshal	Read File
Decision	Check Eligibility
Business Rule (upper)	Update Status

Table 7 Rename Elements

Current Name	Rename to
Business Rule (lower)	Set Bonus
Business Rule	Match Fields
output_Payroll.marshal	Prepare Output
FileClient.write	Process Payroll

Configure Business Rules

Add Business Rules to Links

There are two links in this exercise that use Business Rules applied to links, to move data through the business process model.

1 From the Business Process Designer, select the link between **Receive File** and **Read File**.

2 Right-click on the link and select **Add Business Rule**.

3 Double-click the link with the new Business Rule icon or select the toolbar icon called **Map Business Process Attributes**.

   The Business Rules Designer appears in the lower pane of the Business Process Designer.

4 Link the **text** node to the **contents** node, as shown in **Figure 42 on page 68**.

Figure 42 Add Business Rules to Links
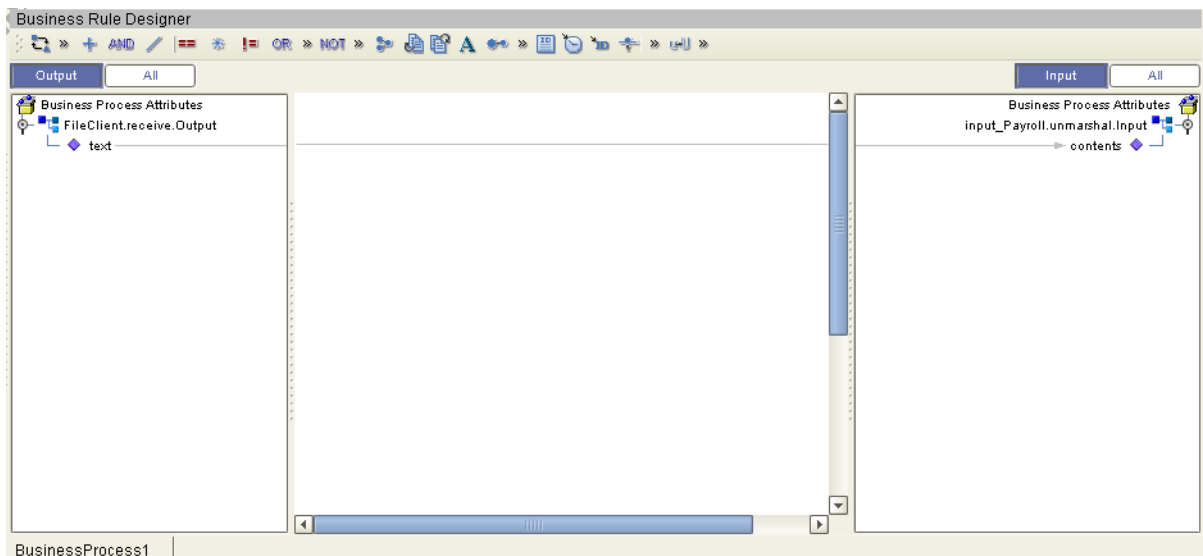
5 Select the link between **Prepare Output** and **Process Payroll**.

6 Right-click on the link and select **Add Business Rule**.

7 Double-click the link with the new Business Rule icon or select the toolbar icon called **Map Business Process Attributes**.

The Business Rules Designer appears in the lower pane of the Business Process Designer.

8 Link the **contents** node to the **text** node.

Decision Element

Configure the Decision Logic

1 Double-click the **Decision** element.

The **Decision Gate Properties** dialog appears, shown in Figure 43.

Figure 43 Decision Properties
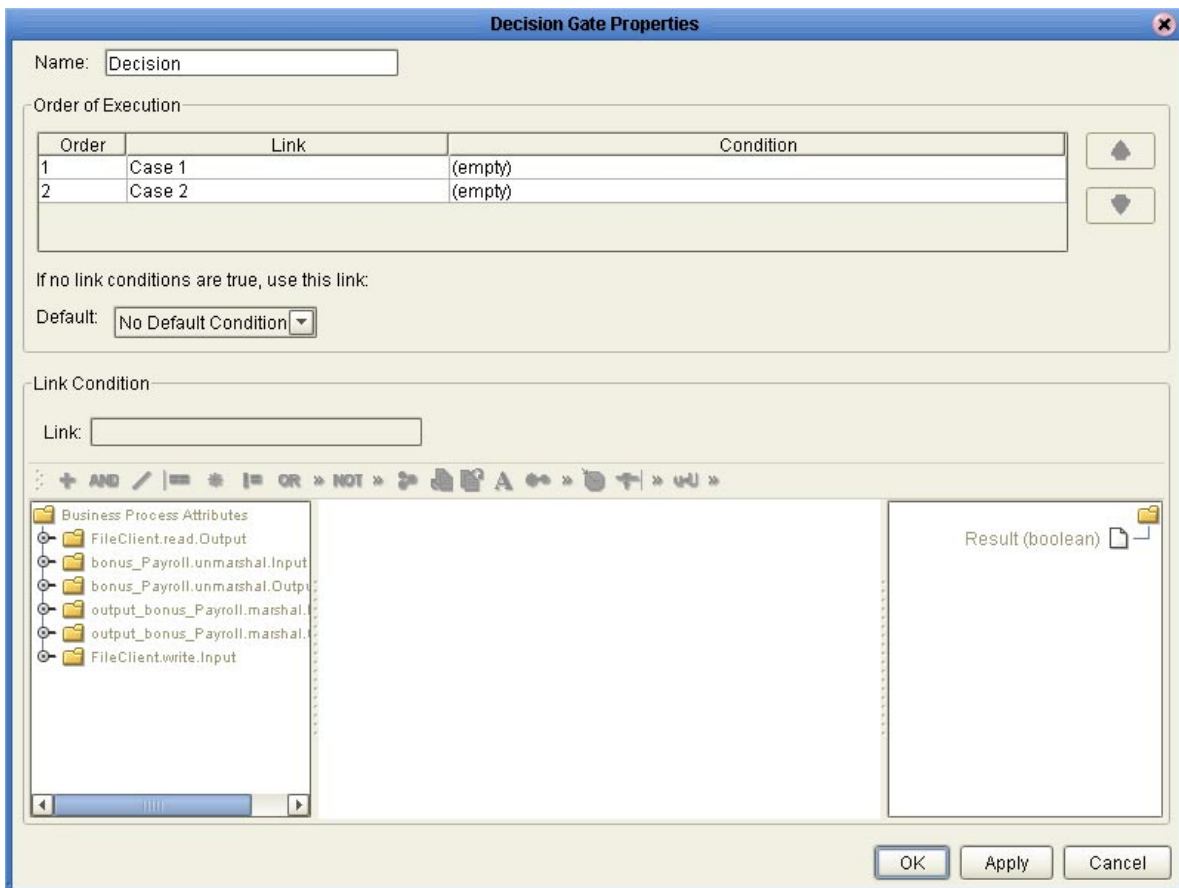
Configure Case 1

2 Select **Case 1** and rename it to **No Bonus**.

3 Select the **String Literal** icon from the **Method Pallet** and drag it to the Business Rules Designer.

The **Input** dialog appears.

4 Type **true** and click **OK**.

5   Select the **equal** method from the Method Pallet and drag it to the Business Rules Designer.

6   In the **Link Condition** section, find the **input_Payroll.unmarshal.Output** node and expand it.

7   Expand the **Payroll** node and select **Probation**.

8   Drag a link from the **Probation** node and connect it the **equal** method box, where you see **Any 1**.

9   Drag a link from the **equal** method box where you see **Any 2** to the **String Literal** method box, and connect.

10 Drag a link from the **Return Boolean** section of the **equal** method box, to the **Result (boolean)** pane on the right.

Your **Decision** mapper should look like **Figure 44 on page 70**.

Figure 44   Completed Decision Gate Properties
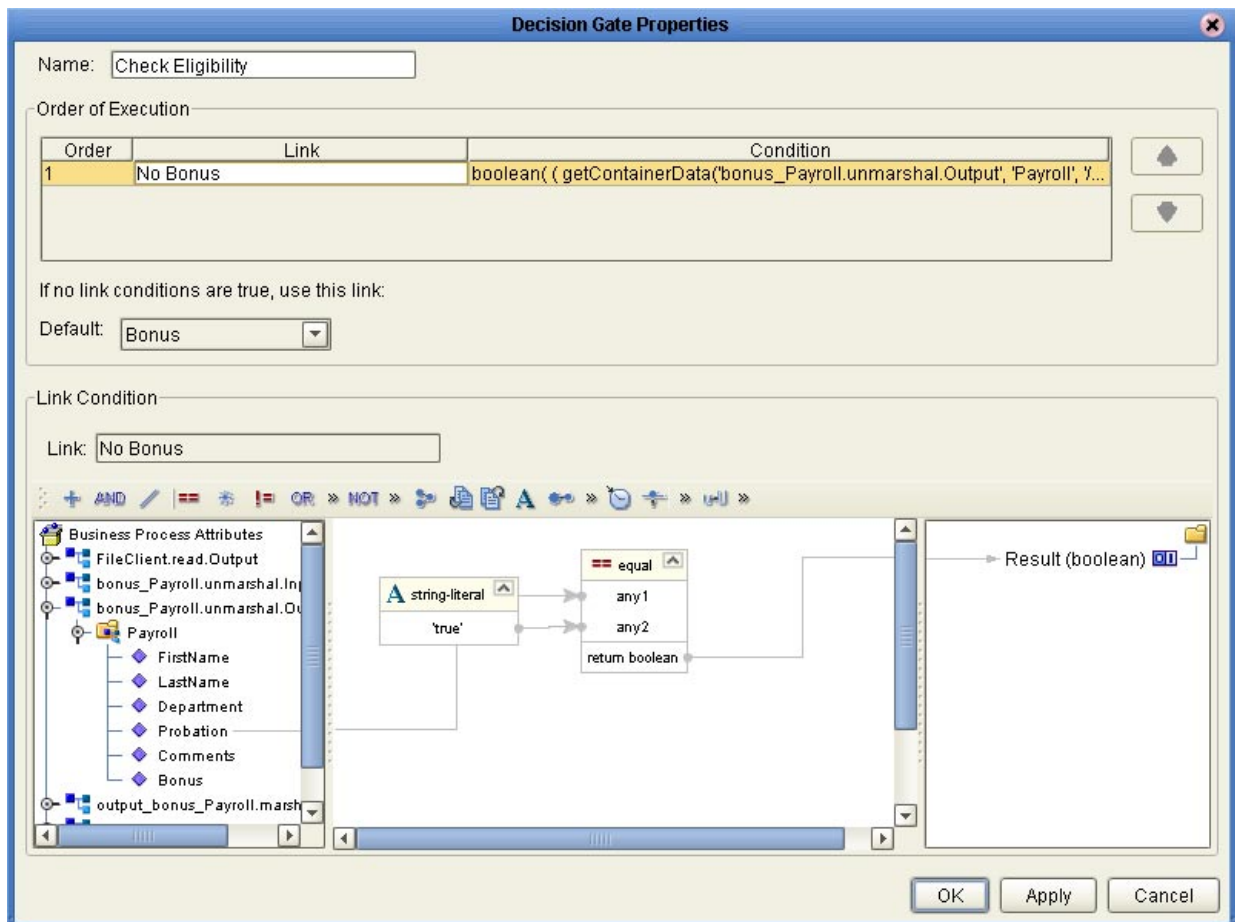
Configure Case 2

1   Select **Case 2** and rename it to **Bonus**.

2   Locate the **Default Condition** drop-down menu and select **Bonus**.

3   Click **OK** to exit the **Decision Gate Properties** dialog.

Your business process model should look like **Figure 41 on page 67**.

6.2.5 Configure Business Rule Activities

Update Status Activity

If an employee is on Probation, they are ineligible for a bonus. In this case, the activity will take the path of the Update Status Activity. The Comments field is set to let the employee know that they are ineligible for a bonus and the bonus amount is set to $0.00.

Update Comments

1   Select the **Update Status** Activity.

2   Click the **Map Business Process Attributes** toolbar icon to see the Business Rules Designer.

3   Drag the **Concat** method onto the Business Rules Designer from the Method Palette.

4   Drag the **String Literal** method onto the Business Rules designer from the Method Palette.

The **Input** dialog appears.

5   Type "**Not Eligible for Bonus**".

6   Link the **String Literal** method to **string1** on the **Concat** method

7   On the left pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

8   Select **Comments** and drag a link to **string2** on the **Concat** method.

9   On the right pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

10 Drag a link from **Return String** on the **Concat** method to **Comments** on the right pane.

Update Bonus

1   Drag the **Concat** method onto the Business Rules Designer from the Method Palette.

2   Drag the **String Literal** method onto the Business Rules designer from the Method Palette.

The **Input** dialog appears.

3   Type "**$0.00**" to set the Bonus amount.

4   Link the **String Literal** method to **string1** on the **Concat** method

5   On the left pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

6   Select **Bonus** and drag a link to **string2** on the **Concat** method.

7 On the right pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

Drag a link from **Return String** on the **Concat** method to **Bonus** on the right pane.

When you are done, your screen should look like Figure 45.

Figure 45   Update Status Activity
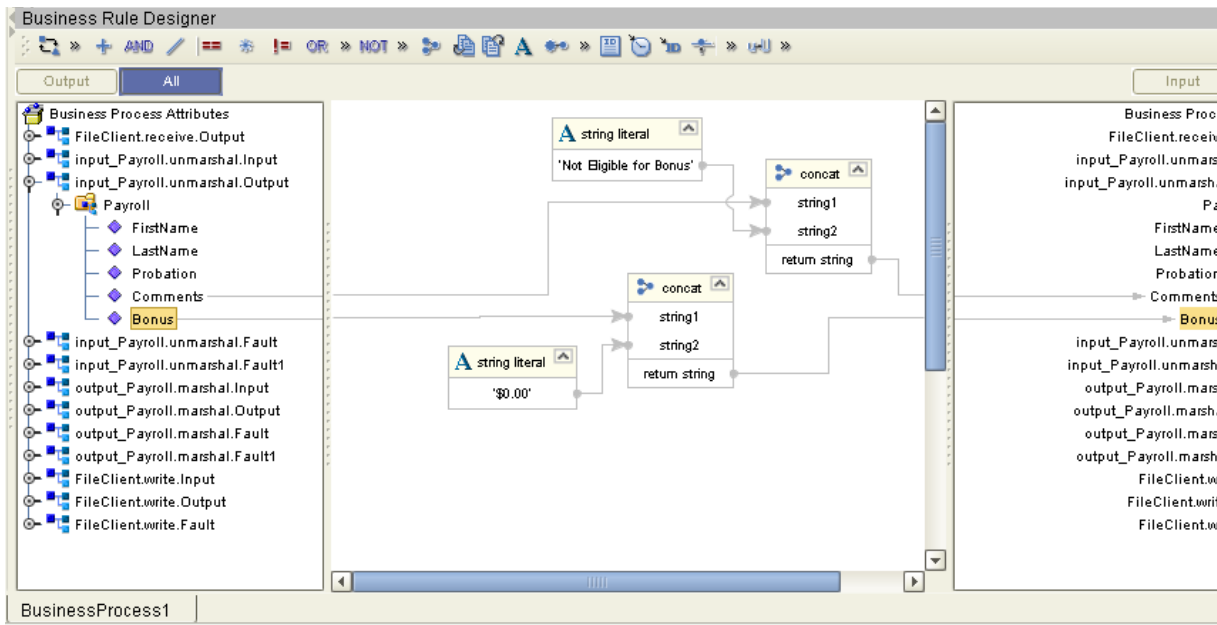
Set Bonus Activity

Update Comments

1 Select the **Set Bonus** Activity.

2 Click the **Map Business Process Attributes** toolbar icon to see the Business Rules Designer.

3 Drag the **Concat** method onto the Business Rules Designer from the Method Palette.

4 Drag the **String Literal** method onto the Business Rules Designer from the Method Palette.

   The **Input** dialog appears.

5 Type "**Eligible for Bonus**".

6 Link the **String Literal** method to **string1** on the Concat method

7 On the left pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

8 Select **Comments** and drag a link to **string2** on the **Concat** Method.

9 On the right pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

10 Drag a link from **Return String** on the **Concat** method to **Comments** on the right pane.

Update Bonus

1 Drag the **Concat** method onto the Business Rules Designer from the Method Palette.

2 Drag the **String Literal** method onto the Business Rules Designer from the Method Palette.

The **Input** dialog appears.

3 Type "**$1500**" to set the bonus that all eligible employees will receive.

4 Link the **String Literal** method to **string1** on the Concat method

5 On the left pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

6 Select **Bonus** and drag a link to **string2** on the **Concat** Method.

7 On the right pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

Drag a link from **Return String** on the **Concat** method to **Bonus** on the right pane.

When you are done, your screen should look like Figure 46.

Figure 46   Set Bonus Activity
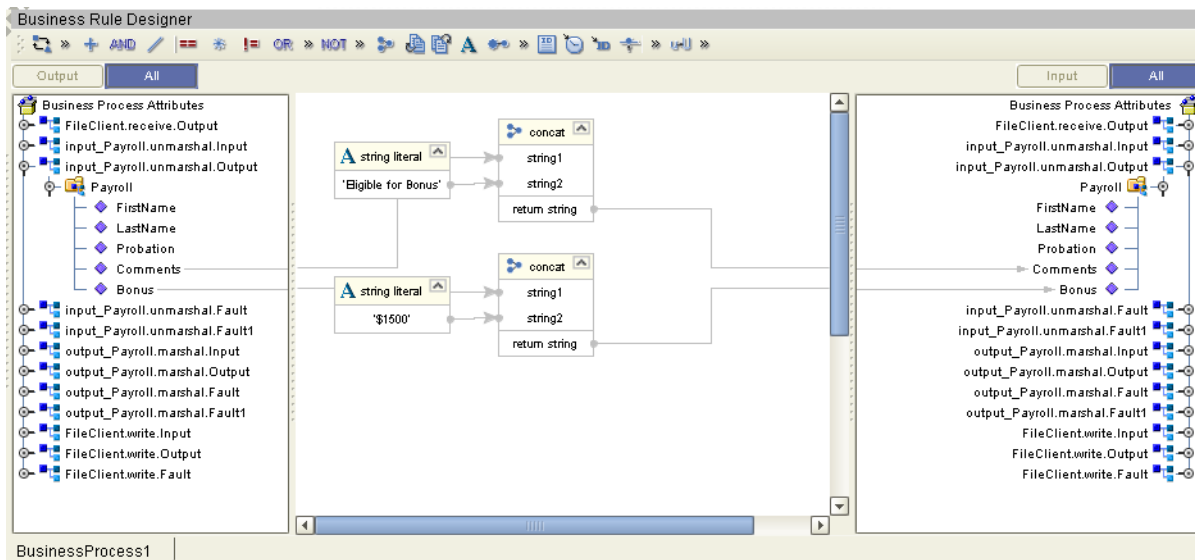
Match Fields Activity

1 Select the **Match Fields** Business Rule Activity.

2 On the left pane, expand the **input_Payroll.unmarshal.Output** node and then expand the **Payroll** node.

3 On the right pane, expand the **output_Payroll.marshal.Input** node and then expand the **Payroll** node.

4 Draw a link between the following fields:

Table 8 Match Fields

From	To
FirstName	FName
LastName	LName
Comments	Message
Bonus	BonusTotal

Your screen will look like Figure 47, when you are finished.

Figure 47 Match Fields Activity
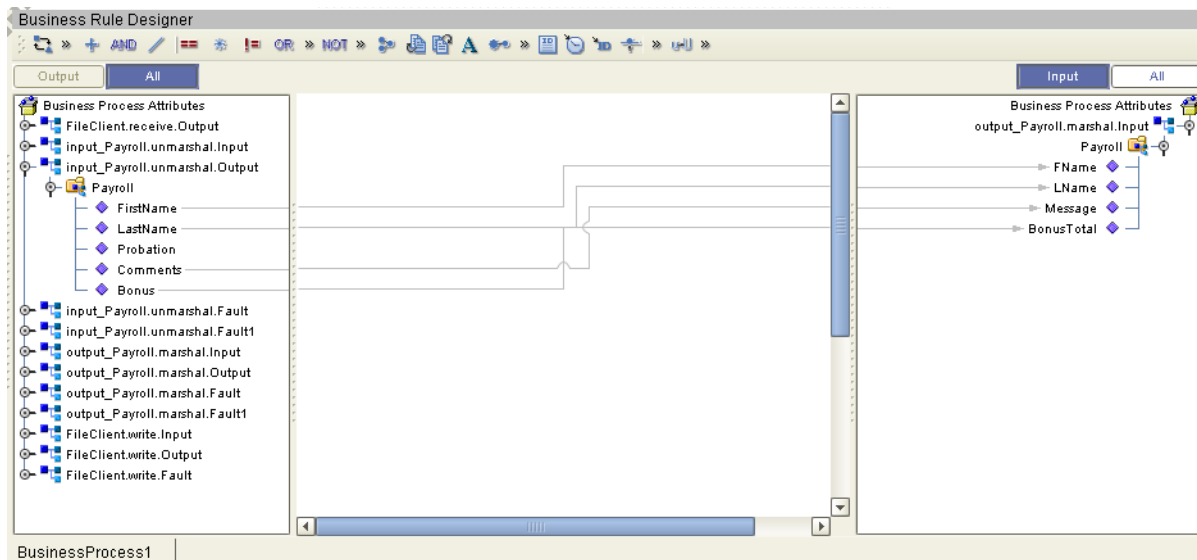
6.2.6 Create a Connectivity Map and Deployment Profile

Create the Connectivity Map

1 Right-click your Project and select **New: Connectivity Map**.

A new node will appear under your Project. The default name is **CMap1**.

2 Select the **External Applications** toolbar icon and select **File External Applications**.

3 Drag the **File** icon to the Connectivity Map canvas.

4 Rename File1 to **File_input**.

5 Drag a second **File** icon to the canvas.

6 Rename the second File to **File_output**.

7 Select **BusinessProcess1** from the Project Explorer and drag it to the canvas.

8 Place the **Business Process** between the two **File** icons.

Configure the Business Process

1 Select your **Business Process** from the Project Explorer and drag it to the canvas.

2 Double-click the **Business Process**.

The **Business Process Service** dialog appears as shown in Figure 48.

3 Drag a link from **File Sender** Service to the **File_input** icon.

4 Drag a link from the **File Receiver** Service to **File_output** icon.

5 Click the minimize button on the **Business Process** dialog to close the dialog.

Figure 48 Configure Binding
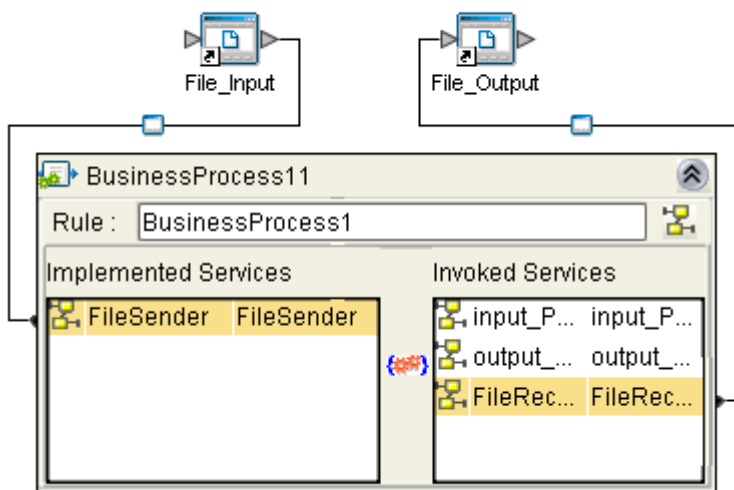
Configure the File Systems

Configure the Inbound File eWay

1 Double-click the link to **File_input** to configure it.

The **Templates** dialog appears.

2 Select Inbound **File eWay** and select **OK**.

The **Properties** dialog appears.

3 Change the **Directory** to **C:\data**.

4 Change **Input** file name to ***.xml**.

5 Click **OK** to save changes.

Configure the Outbound File eWay

1 Double-click the link **File_output** to configure it.

2 Connect File Receiver to **File_output**.

The **Templates** dialog appears.

3 Select **Outbound File eWay** and select **OK**.

The **Properties** dialog appears.

4 Change the **Directory** to **C:\data**.

5 Change the **Output** file name to **output%d.txt.**

6 Click **OK** to save changes.

Deployment Profile

Create the Deployment Profile

1 Right-click your **Project** from the **Project Explorer**.

2 Select **New: Deployment Profile**.

3 The **Create Deployment Profile** dialog appears.

4 The **Deployment Profile** is called **Deployment1** by default. You can accept the default name.

5 Select the **Environment** (**Environment1**) that you created previously.

Configure the Deployment Profile

1 Drag **BusinessProcess11** from the middle pane to the **Integration Server** (**IntegrationSvr1**) located in the **LogicalHost** window.

2 Drag **File1 -> BusinessProcess11** from the middle pane to the **Pay_IN** window.

3 Drag **BusinessProcess11 -> File2** to the **Pay_OUT** window.

4 Click **Activate**.

Figure 49 Deployment Profile
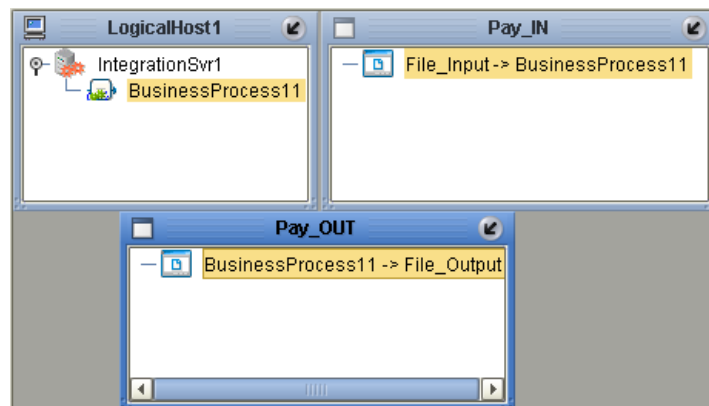
5 Click **No**, so the change will not apply to the **Logical Host** immediately. (This is because your Logical Host is not running yet.)

6.2.7 **Run and Test the System**

Start the Logical Host

The Bootstrap process executes your **Payroll Project** and begins the process of polling your input data. The Bootstrap process is performed from a command prompt. Bootstrap will pick up the deployment profile the first time it runs; after that you would select reactivate and click **Yes** to apply the most recent changes to the Logical Host.

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

To run the Bootstrap

1 Open a Windows command prompt as shown in the following Figure 50.

2 Navigate to where you installed the Logical Host; for example, **ican50\logicalhost\bootstrap\bin**, then type the following command:

```
CD \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
```

environment_name is the name of your environment (for example, *Environment1*),
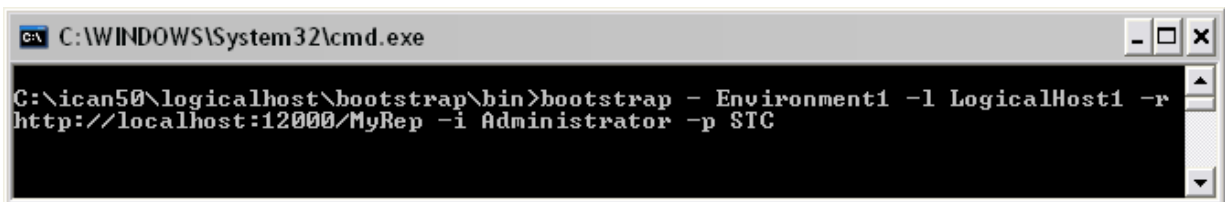
logicalhost_name is the name of your Logical Host (for example, *LogicalHost1*),

repository_URL is the full URL of your Repository including the Repository name (for example, *http://localhost:12000/MyRep*),

username is your user name, and

password is your password.

Figure 50 Bootstrap Command Example

Check output

1 Navigate to **C:\data** and check for an **output.txt** file.

2 If the file exists, open it and examine the data. It should look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<Payroll>
  <FName>Tonya</FName>
  <LName>Lee</LName>
  <Message>Not Eligible for Bonus</Message>
  <BonusTotal>$0.00</BonusTotal>
</Payroll>
<?xml version="1.0" encoding="UTF-8"?>
<Payroll>
```

```
        <FName>Vanessa</FName>
        <LName>Smith</LName>
        <Message>Eligible for Bonus</Message>
        <BonusTotal>$1500</BonusTotal>
    </Payroll>
```

Monitoring

See **"Persistence and Monitoring" on page 52** for details about setting up your Business Process for Monitoring.

Appendix A

Method Palette

7.3 Operators

Figure 51 Method Palette: Operator tab
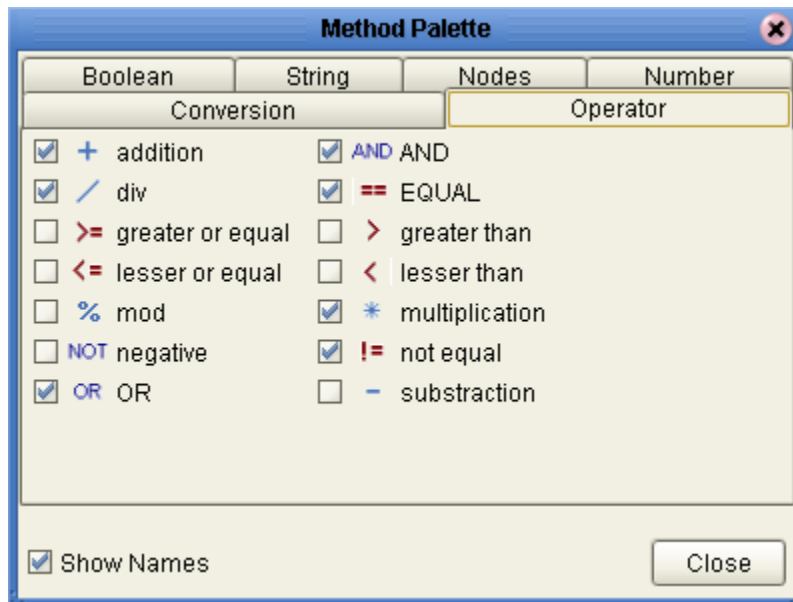
Table 9 Operator Methods

Method Box	Name	Description/Usage
	addition	Adds the value of *number1* to the value of *number2*, returns the sum.

Table 9   Operator Methods (Continued)
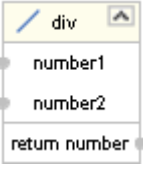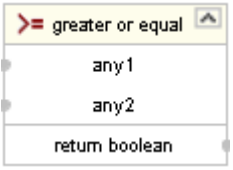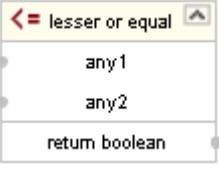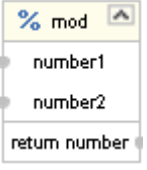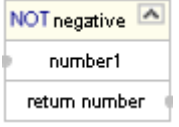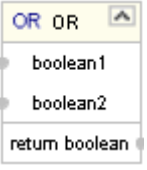
Method Box	Name	Description/Usage
	division	Divides the value of *number1* by the value of *number2*, returns the quotient.
	greater_or_equal	Returns Boolean true if *number1* is greater than or equal to *number2*; otherwise, returns Boolean false.
	lesser_or_equal	Returns Boolean true if *number1* is less than or equal to *number2*; otherwise, returns Boolean false.
	mod	
	negative	Converts the input number to negative. Result is a negative number having the same absolute value as the input number.
	or	Returns Boolean false if both *boolean1* and *boolean2* are false; otherwise, returns Boolean true.

Table 9   Operator Methods (Continued)
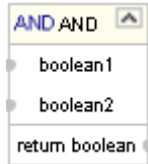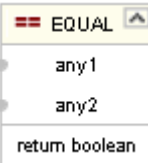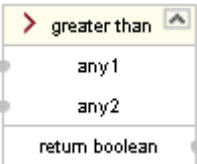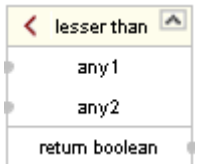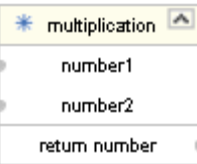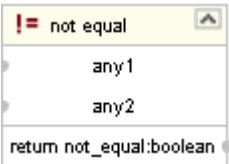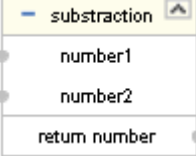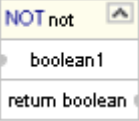
Method Box	Name	Description/Usage
	and	Returns Boolean true if both *boolean1* and *boolean2* are true; otherwise, returns Boolean false.
	equal	Returns Boolean true if *number1* is equal to *number2*; otherwise, returns Boolean false.
	greater_than	Returns Boolean true if *number1* is greater than *number2*; otherwise, returns Boolean false.
	less_than	Returns Boolean true if *number1* is less than *number2*; otherwise, returns Boolean false.
	multiplication	Multiplies the value of *number1* by the value of *number2*, returns the product.
	not_equal	Returns Boolean true if *number1* is not equal to *number2*; otherwise, returns Boolean false.

Table 9   Operator Methods (Continued)

Method Box	Name	Description/Usage
substraction / number1 / number2 / return number	subtraction	Subtracts the numerical value of *number2* from the numerical value of *number1,* returns the difference.
NOT not / boolean1 / return boolean	not	Returns the inverse of *boolean1*.

7.4   String

Figure 52   Method Palette: String tab
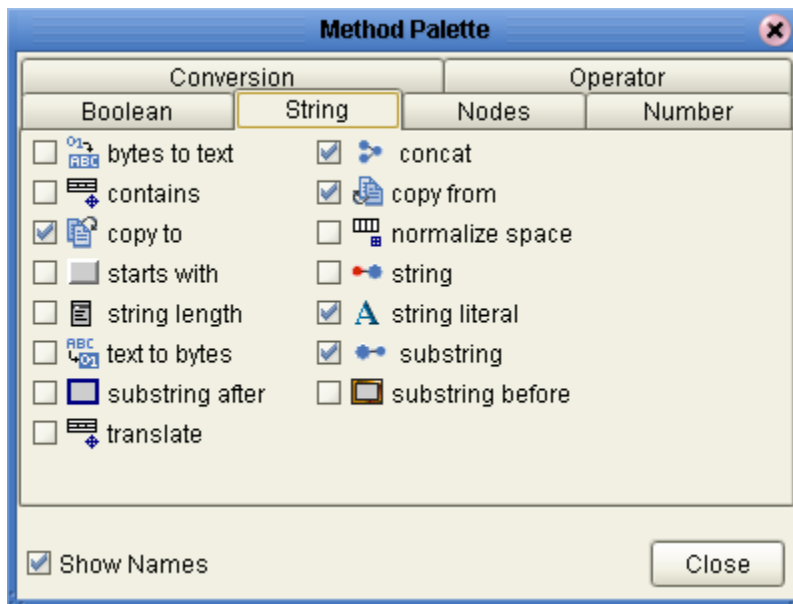
Table 10   **String Methods**
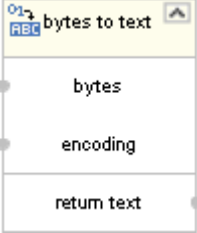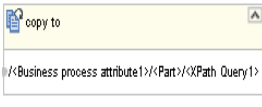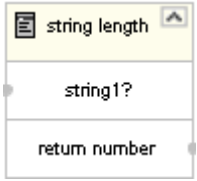
Symbol	Name	Description
bytes to text bytes encoding return text	bytes to text	Decodes bytes into text using the specified encoding. If no encoding is specified, the platform's default encoding is used.
contains string1 string2 return boolean	contains	Returns true if the second string is contained within the first string, otherwise it returns false
copy to /<Business process attribute1>/<Part>/<XPath Query1>	copy to	Allows you to type in the xpath expression for the destination of a copy operation. This is useful for entering xpath predicates. Note: This is for advanced users who are familiar with xpath and BPEL syntax.
starts with string1 string2 return boolean	starts with	Returns true if the first string starts with the second string, otherwise it returns false
string length string1? return number	string length	Returns the number of characters in a string

Table 10   String Methods (Continued)
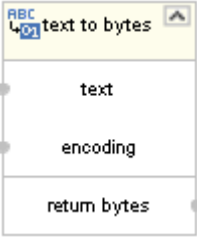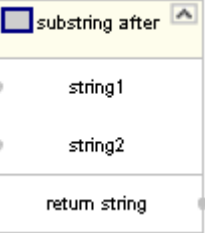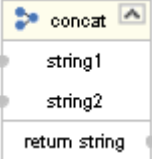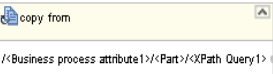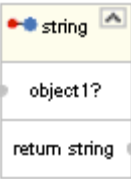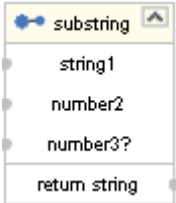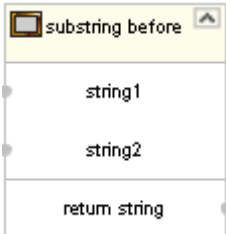
Symbol	Name	Description
	text to bytes	Encodes the input text into a sequence of bytes using the specified encoding. If no encoding is specified, the platform's default encoding is used
	substring after	Returns the part of the string in the string argument that occurs after the substring in the substring argument
	translate	Performs a character by character replacement. It looks in the value argument for characters contained in string1, and replaces each character for the one in the same position in the string2
	concat	Returns the concatenation of all its arguments
	copy from	Allows you to type in xpath expression for the source of a copy operation. This is useful for entering xpath predicates. Note: This is for advanced users who are familiar with xpath and BPEL syntax

Table 10   String Methods (Continued)

Symbol	Name	Description
normalize space / string1? / return string	normalize space	Removes leading and trailing spaces from a string
string / object1? / return string	string	Converts the value argument to a string
A string literal / Lit1	string literal	A sequence of characters of fixed length and content
substring / string1 / number2 / number3? / return string	substring	Returns a part of the string in the string argument
substring before / string1 / string2 / return string	substring before	Returns the part of the string in the string argument that occurs before the substring in the substring argument.

7.5   Number

Figure 53   Method Palette: Number tab
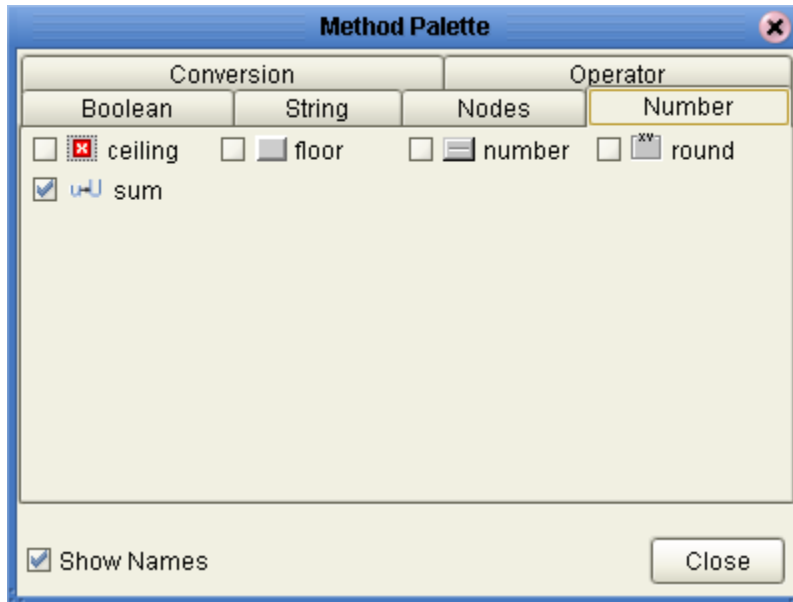
Table 11   **Number Methods**
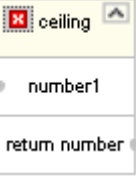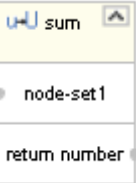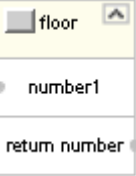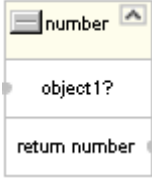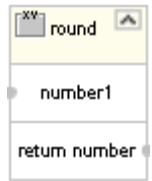
Symbol	Name	Function
	ceiling	Returns the smallest integer that is not less than the number argument
	sum	Returns the total value of a set of numeric values in a node-set
	floor	Returns the largest integer that is not greater than the number argument

Table 11   Number Methods (Continued)

Symbol	Name	Function
	number	Converts the value argument to a number
	round	Rounds the number argument to the nearest integer

7.6   Boolean

Figure 54   Method Palette: Boolean tab
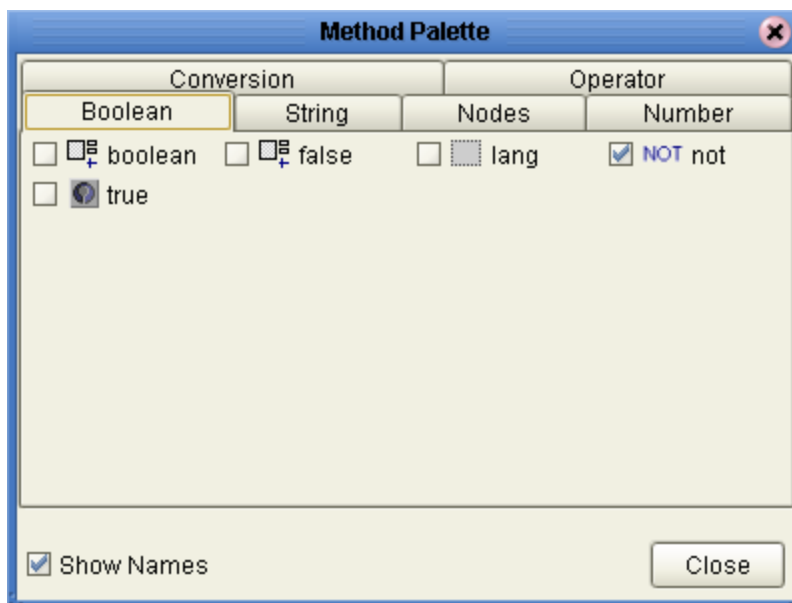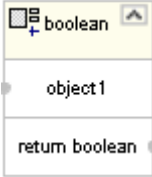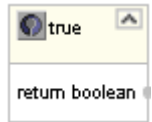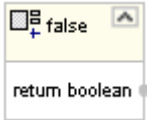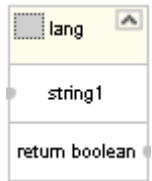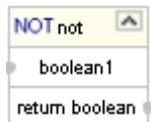
Table 12   Boolean Methods

Symbol	Name	Function
	boolean	Converts the value argument to Boolean and returns true or false
	true	Returns true
	false	Returns false
	lang	Returns true if the language argument matches the language of the xsl:lang element, otherwise it returns false
	not	Returns true if the condition argument is false, and false if the condition argument is true

7.7   Nodes

Figure 55   Method Palette: Nodes tab
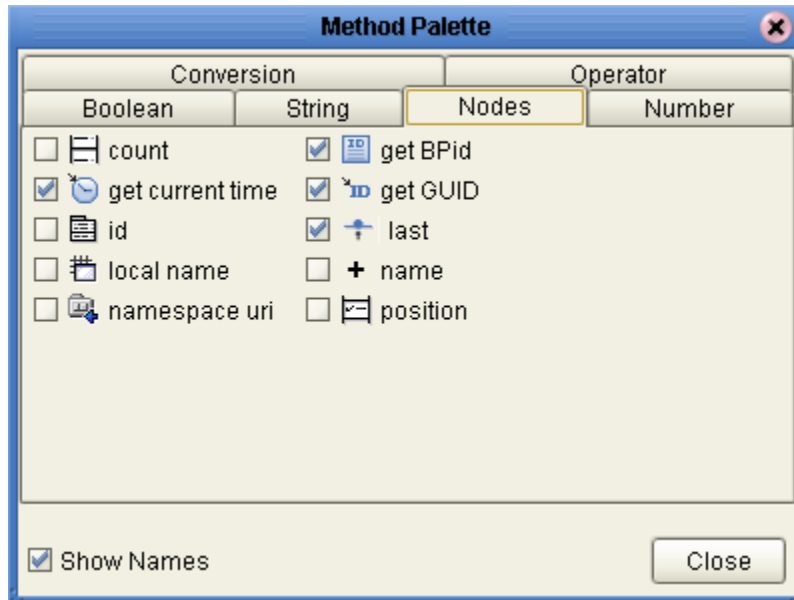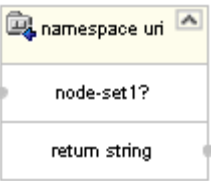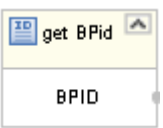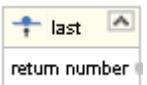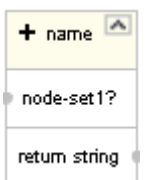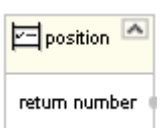
Table 13   **Nodes Methods**

Symbol	Name	Function
	count	Returns the number of nodes in a node-set
	getCurrentTime	Gets the current time in ISO 8601 format (e.g. 2003-08-15T02:03:49.92Z).
	id	Selects elements by their unique ID

Table 13 Nodes Methods (Continued)

Symbol	Name	Function
local name · node-set1? · return string	local name	Returns the local part of a node. A node usually consists of a prefix, a colon, followed by the local name
namespace uri · node-set1? · return string	namespace uri	Returns the namespace URI of a specified node
get BPid · BPID	getBPId	Gets the business process instance ID.
get GUID · GUID	getGUId	Gets a randomly generated globally unique ID.
last · return number	last	Returns the position number of the last node in the processed node list
name · node-set1? · return string	name	Returns the name of a node
position · return number	position	Returns the position in the node list of the node that is currently being processed

Glossary

activity
> An organizational unit for performing a specific function. An activity defines a step of a particular business process.

activity states
> The stages that activities within the business process instance go through as the business process version is being run.

Business Process Attribute
> Attributes pass user-defined control information (programming arguments) to and from the eInsight Business Process Manager and its activities.

business process
> A business process is a collection of actions and messages, revolving around a specific business practice, that flow in a specific pattern to produce an end result.

business process attributes
> Business process attributes pass user-defined control information (programming arguments) to and from the e*Insight Business Process Manager, external sources, and internal components.

business process instance (BPI)
> A unique instantiation of a business process.

business process model
> The graphical representation of a business process.

business process version
> A form or variant of the original business process model.

Collaboration
> A component of an eWay that receives and processes Events and forwards the output to other eGate components.

eInsight Business Process Manager (eInsight)
> The component within the ICAN product suite that facilitates the automation of the business process flow of business activities.

Decision
> Controls the logical flow of data-based decisions in the business process model. A Decision outputs specific information when specified input conditions are met.

GUI

Graphical User Interface. A type of computer interface that enables the user to perform actions via the use of symbols, visual metaphors and pointing devices.

Business Process Designer

The Business Process Designer is the portion of the eInsight where you create the business process model, in the form of a flow chart.

security

Security is the ability to limit user access to specific items based on a pre-determined profile of the user.

state

See *activity states*

string

A sequence of text characters.

Sub-process

A sub-process is a business process which is called, or used by, another business process, as a sub-component.

tree view

The tree view displays a hierarchical representation of all the components, and their activities.

User activity

Allows external applications to access attributes in the business process.

Index