

SeeBeyond ICAN Suite

eGate Integrator for eInsight Enterprise Service Bus User's Guide

Release 5.0



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

SeeBeyond, e*Gate, and e*Way are the registered trademarks of SeeBeyond Technology Corporation in the United States and select foreign countries; the SeeBeyond logo, e*Insight, and e*Xchange are trademarks of SeeBeyond Technology Corporation. The absence of a trademark from this list does not constitute a waiver of SeeBeyond Technology Corporation's intellectual property rights concerning that trademark. This document may contain references to other company, brand, and product names. These company, brand, and product names are used herein for identification purposes only and may be the trademarks of their respective owners.

© 2003 by SeeBeyond Technology Corporation. All Rights Reserved. This work is protected as an unpublished work under the copyright laws.

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

Version 20031014212047.

Contents

List of Figures	9
List of Tables	14
<hr/>	
Chapter 1	
Introduction	15
Purpose and Scope	15
Intended Audience	15
Organization of Information	15
Writing Conventions	16
Supporting Documents	17
Online Documents	17
The SeeBeyond Web Site	17
<hr/>	
Chapter 2	
System Overview	18
Introduction	18
Integration Model	19
System Architecture	21
Repository	22
Environments	22
User Interfaces	23
Enterprise Designer	23
Enterprise Manager	25
<hr/>	
Chapter 3	
Enterprise Designer	26
Overview	26

Menu Bar	28
File Menu	28
Tools Menu	28
View Menu	28
Window Menu	29
Help Menu	29
Toolbar	30
Browser Buttons	30
Enterprise Explorer	31
Project Explorer	31
Project Explorer Icons	32
Context Menus	33
Repository Menu	33
Project Menu	34
Connectivity Map Menu	35
Environment Explorer	36
Environment Explorer Icons	37
Context Menus	38
Repository Menu	38
Environment Menu	39
Logical Host Menu	40
Enterprise Designer Editors	42
Connectivity Map Editor	43
OTD Editor	44
Environment Editor	45
Deployment Editor	46

Chapter 4

Creating Projects	47
Overview	47
Project Components	48
Starting Enterprise Designer	49
Using the Project Explorer	51
Using the Connectivity Map Editor	52
Connectivity Map Editor Toolbar	53
Creating and Configuring Logical Components	54
Services	54
External Applications	54
Schedulers	55
Component Connections	56
Configuring a Connection	57
Modifying a Configuration Property	59
Defining Constants and Variables	60
Web Services	63

Example Web Services Project	65
Creating the OTDs	65
Using the Connectivity Map	67

Chapter 5

Object Type Definitions	69
Overview	69
OTD Types	70
Externally-Defined OTDs	70
Building OTDs	70
Using the OTD Wizard	71
Navigation Buttons	72
WSDL-Based OTDs	73
Using the WSDL Wizard	73
WSDL OTD Operation	77
WSDL Operation Elements	77
WSDL OTD Structure	77
XSD-Based OTDs	78
Using the XSD Wizard	78
Using the OTD Editor	82
Node Management	84
Using the OTD Tester	85

Chapter 6

Environments and Project Deployment	88
Overview	88
Environmental Constants	90
Deployment Profiles	91
Using the Deployment Editor	92
Creating a Deployment Profile	93
Activating and Deactivating Deployments	96
Mapping Variables	98
Deploying Projects to Third-Party Servers	99
BEA WebLogic	99
IBM WebSphere	102

Chapter 7

Logical Hosts	105
Overview	105
Integration Servers	105
Message Servers	105
Management Agent	105
Bootstrap	106
Integration Servers	106
Configuring a Logical Host	107
Logical Host Startup Parameters	107
Modifying the Logical Host Startup Configuration File	108
Configuring the Base Port Number	111
Starting the Logical Host	112
Starting the Logical Host as a Windows Service	113
Starting the Logical Host Manually on a Windows System	114
Starting the Logical Host on a UNIX System	115
Starting the Logical Host on a Red Hat Linux System	115
Starting the Logical Host on an HP NonStop Server	116

Chapter 8

Repository Tools	117
Overview	117
Impact Analysis	118
Command Buttons	119
Using the Impact Analyzer	119
Printing Impact Analysis Results	120
Version Control	121
Viewing History of a Project Component	121
Checking In a Project or Environment Component	122
Checking Out a Project or Environment Component	123
Repository Backup and Restoration	124
Backing Up a Repository	124
Restoring a Repository	124
Project Export and Import	126
Exporting a Project	126
Enterprise Designer Method	126
Command-Line Method	127
Importing a Project	128
Enterprise Designer Method	128
Command-Line Method	129

Chapter 9

Security	130
Overview	130
Configuration User Management	131
Roles	131
User Management Interface	132
Environment User Management	134
Message Server Security	135
Message Server Settings	135
JMS Client Security	135
ACL Management	136
SSL/HTTPS	138
Overview	138
Certificates and Keys	138
The Keytool Utility	138
Installation and Configuration	139
LDAP	142
Overview	142
Referencing and Accessing Information	143
Implementing LDAP	144
Configuring the ICAN Suite to Use LDAP	144
Configuration User Management	144
Runtime User Management	145

Chapter 10

Enterprise Manager	146
Overview	146
Starting Enterprise Manager	147
Home	149
Monitor Server	149
Administration	153
Downloads	156
Documentation	157

Chapter 11

Managing Projects	158
Overview	158
Accessing the Monitor Server	159

Monitoring Collaborations	160
Viewing Alerts	166
Viewing Logs	168
Integration Server Level	168
Collaboration Level	169
Setting Log Levels	170
Indoubt Transaction Editing	171

Chapter 12

Logging	173
Logging	173
eGate Logs	174
Log File System	174
Log File Management	174
Log File Properties	175
Logical Host	175
Integration Server	175
Log Files and Locations	176
Repository	176
Logical Host	177
Management Agent	177
Integration Server	177
Message Server	177
Enterprise Designer	177
Glossary	178
e*Gate 4.x Terms in eGate 5.0	181
Index	183

List of Figures

Figure 1	SeeBeyond ICAN Suite	18
Figure 2	eGate Integrator	19
Figure 3	eGate Integrator Implementation Model	20
Figure 4	Typical eGate Integrator System	21
Figure 5	Enterprise Designer	23
Figure 6	Connectivity Map Editor	24
Figure 7	SeeBeyond Enterprise Manager Login	25
Figure 8	SeeBeyond Enterprise Designer	26
Figure 9	Enterprise Explorer: Project Explorer View	31
Figure 10	Repository Menu	33
Figure 11	Project Menu	34
Figure 12	Connectivity Map Menu	35
Figure 13	Enterprise Explorer: Environment Explorer View	36
Figure 14	Repository Menu	38
Figure 15	Environment Menu	39
Figure 16	Logical Host Menu	40
Figure 17	Connectivity Map Editor	43
Figure 18	OTD Editor	44
Figure 19	Environment Editor	45
Figure 20	Deployment Editor	46
Figure 21	Login Dialog Box	49
Figure 22	SeeBeyond Enterprise Designer	50
Figure 23	Project Explorer	51
Figure 24	Connectivity Map Window	52
Figure 25	Linking JMS Topics	52
Figure 26	Service Component	54
Figure 27	External Application Drop-Down Menu	54
Figure 28	Selected External Applications in Toolbar	55
Figure 29	Scheduler Component	55
Figure 30	Connection Icons in a Connectivity Map	56
Figure 31	Default Configuration Dialog Box	57
Figure 32	Editing a Default Configuration Field	59

List of Figures

Figure 33	Project Variable Creation	60
Figure 34	Project Constant Creation	61
Figure 35	Variables and Constants Object Group	61
Figure 36	Connector Properties	62
Figure 37	SeeBeyond UDDI Repository	63
Figure 38	Example Web Service WSDL File	63
Figure 39	Microsoft Visual Studio Example	64
Figure 40	Web Service Example (OTD 1)	65
Figure 41	Web Service Example (OTD 2)	65
Figure 42	Web Service Example (OTD 3)	66
Figure 43	Web Service Example (Connectivity Map)	67
Figure 44	Web Service Example (WSDL File)	68
Figure 45	Web Service Example (UDDI Repository)	68
Figure 46	OTD Wizard Selection Dialog	71
Figure 47	OTD Wizard Selection: WSDL Wizard	73
Figure 48	WSDL Wizard: Select WSDL Location	74
Figure 49	WSDL Wizard: Select WSDL File	75
Figure 50	WSDL Wizard: Select OTD Options	76
Figure 51	OTD Wizard Selection: XSD Wizard	78
Figure 52	XSD Wizard: Select XSD File(s)	79
Figure 53	Select Document Elements Dialog Box	80
Figure 54	Select OTD Options Dialog Box	81
Figure 55	OTD Editor	82
Figure 56	OTD Tester	85
Figure 57	Test Panel Data Display	85
Figure 58	Select Data File	86
Figure 59	Object Elements and Values	86
Figure 60	Data Display: Refresh Icon	87
Figure 61	Status Data Display	87
Figure 62	Environment Editor	88
Figure 63	Repository Context Menu	89
Figure 64	Environment Context Menu	89
Figure 65	Logical Host Context Menu	89
Figure 66	Environmental Constants Editor	90
Figure 67	eGate Integrator Implementation Model	91
Figure 68	Deployment Editor Window	92
Figure 69	Example Deployment Profile (1)	93
Figure 70	Example Deployment Profile (2)	94

List of Figures

Figure 71	Example Deployment Profile (3)	95
Figure 72	Activate Dialog Box	96
Figure 73	Logical Host Context Menu - Apply	96
Figure 74	Activate Dialog Box	97
Figure 75	Deployment Profile Mappings	98
Figure 76	Project Variable Value Entry	98
Figure 77	WebLogic Deployment (1)	99
Figure 78	WebLogic Deployment (2)	100
Figure 79	WebLogic Deployment Verification	101
Figure 80	WebSphere Deployment (1)	102
Figure 81	WebSphere Deployment (2)	103
Figure 82	WebSphere Deployment Verification	104
Figure 83	Integration Server (J2EE Compatible)	106
Figure 84	Example logical-host.properties File	108
Figure 85	Logical Host Properties Dialog Box	111
Figure 86	Logical Hosts	112
Figure 87	Install as Service Script	113
Figure 88	Windows Services List	113
Figure 89	Uninstall as Service Script	114
Figure 90	Impact Analyzer Dialog Box	118
Figure 91	Print Dialog Box	120
Figure 92	Printed Component Data	120
Figure 93	Version Control - History Dialog Box	121
Figure 94	Version Control Dialog Box	122
Figure 95	Version Control - Check Out Dialog Box	123
Figure 96	Enter File Name Dialog Box	126
Figure 97	Message Dialog Box	127
Figure 98	Select File Dialog Box	128
Figure 99	File Destination Dialog Box	129
Figure 100	Message Dialog Box	129
Figure 101	User Management Dialog Box	132
Figure 102	User Management - Add User	133
Figure 103	Environment Context Menu	134
Figure 104	JMS IQ Manager Properties	135
Figure 105	JMS Client Security Properties	135
Figure 106	ACL Management Dialog Box	136
Figure 107	ACL Add Users Dialog Box	137
Figure 108	ACL Management Dialog Box - Read and Write Access	137

List of Figures

Figure 109	LDAP Directory Tree (Traditional Naming)	142
Figure 110	LDAP Directory Tree (Internet Naming)	143
Figure 111	Security Configuration Template Properties Dialog Box	145
Figure 112	Enterprise Manager Login	147
Figure 113	Enterprise Manager GUI	147
Figure 114	Monitor Launch Window	149
Figure 115	Monitor Page - Projects	150
Figure 116	Monitor Page - Environment - Alerts	151
Figure 117	Monitor Page - Environment - List	151
Figure 118	Monitor Page - Environment - Log Props	152
Figure 119	Upload Product License	153
Figure 120	Upload eGate SAR File	153
Figure 121	Upload Product Manifest	154
Figure 122	Choose File Dialog Box	154
Figure 123	Products Available to Upload Page	155
Figure 124	Products Available to Download Page	156
Figure 125	Documentation Tab	157
Figure 126	Monitor Launch Window	159
Figure 127	Example Project Connectivity Map	160
Figure 128	Monitor - Details - List Collaborations	160
Figure 129	Starting and Stopping Collaborations - Details Panel	161
Figure 130	Starting and Stopping Collaborations - Explorer	161
Figure 131	Pending and Processed Messages	162
Figure 132	Monitor - Project Explorer	163
Figure 133	Active/Non-active Collaborations	163
Figure 134	Pending and Processed Messages	164
Figure 135	Logging Details	164
Figure 136	Connectivity Map Details: Components	165
Figure 137	Connectivity Map View - Zoom In/Out	165
Figure 138	Collaboration Alerts	166
Figure 139	Alert Status	166
Figure 140	Alert Details	167
Figure 141	Integration Server Log Messages	168
Figure 142	Integration Server Log Messages - Filtered	168
Figure 143	Collaboration Log File	169
Figure 144	Search on Keyword	169
Figure 145	Logging Properties Page	170
Figure 146	Resetting Log Levels	170

List of Figures

Figure 147	Message Server Details - Controls Tab	171
Figure 148	Indoubt Transaction List	172
Figure 149	Indoubt Transaction List - Transaction Selected	172
Figure 150	Recirculating Log File Stack	174

List of Tables

Table 1	Writing Conventions	16
Table 2	File Menu Options	28
Table 3	Tools Menu Options	28
Table 4	View Menu Options	28
Table 5	Window Menu Options	29
Table 6	Help Menu Options	29
Table 7	Enterprise Designer Toolbar Icons	30
Table 8	Browser Buttons	30
Table 9	Project Icons	32
Table 10	Repository Menu Options	33
Table 11	Project Menu Options	34
Table 12	Connectivity Map Menu Options	35
Table 13	Environment Icons	37
Table 14	Repository Menu Options	38
Table 15	Environment Menu Options	39
Table 16	Logical Host Menu Options	40
Table 17	Connectivity Map Toolbar Icons	53
Table 18	Default Configuration Toolbar Buttons	58
Table 19	OTD Wizard Navigation Buttons	72
Table 20	OTD Editor Toolbar Icons	83
Table 21	Environmental Constants Editor Icons	91
Table 22	Deployment Toolbar Buttons	92
Table 23	Command Arguments for bootstrap	107
Table 24	Logical Host Properties file	110
Table 25	Navigation Buttons	119
Table 26	Realm Element Attributes	144
Table 27	Log Properties for Logical Hosts	175
Table 28	Log Properties for Integration Servers	175
Table 29	eGate 5.0 Terms	181

Introduction

This chapter introduces you to this *eGate Integrator User's Guide*, its general purpose and scope, and its organization. It also provides sources of related documentation and information.

1.1 Purpose and Scope

The *eGate Integrator User's Guide* provides general information about the features and operation of SeeBeyond® eGate Integrator 5.0.

Note: *Any operation explanations provided in this document are generic, for reference purposes only, and do not necessarily address the specifics of setting up individual eGate Projects.*

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 eGate will be installed (Windows or UNIX) and must be thoroughly familiar with Windows-style GUI operations.

1.3 Organization of Information

This document provides information about eGate Integrator 5.0 and includes the following chapters:

- **Chapter 1 “Introduction”** describes the purpose of *eGate Integrator User's Guide* includes writing conventions and a list of related documents.
- **Chapter 2 “System Overview”** provides an overview of the general structure, architecture, and operation of the eGate system.
- **Chapter 3 “Enterprise Designer”** provides a detailed overview of the Enterprise Designer, including its structure and operation.

- **Chapter 4 “Creating Projects”** explains how to create a Connectivity Map and use the Configuration Editor to modify eWay and JMS connections between Connectivity Map components.
 - **Chapter 5 “Object Type Definitions”** describes how to create Object Type Definitions (OTDs).
 - **Chapter 6 “Environments and Project Deployment”** explains how to create and activate deployment profiles.
 - **Chapter 7 “Logical Hosts”** explains how to configure and start Logical Hosts.
 - **Chapter 8 “Repository Tools”** looks at the Project analysis tools available in the Enterprise Designer.
 - **Chapter 9 “Security”** discussed the various security features in the ICAN Suite.
 - **Chapter 10 “Enterprise Manager”** provides a detailed overview of the Enterprise Manager, including its structure and operation.
 - **Chapter 11 “Managing Projects”** describes the eGate management and monitoring tools available with the Enterprise Manager.
 - **Chapter 12 “Logging”** provides troubleshooting tips and describes the eGate logs
- In addition, refer to the [Glossary](#) on page 178 for a list of eGate-related terminology.

1.4 Writing Conventions

The writing conventions listed in this section are observed throughout this document.

Table 1 Writing Conventions

Text	Convention	Example
Buttons, File, Icon, and Menu Names	Bold text	Click OK to save and close. From the File menu, select Exit . Select the logicalhost.exe file.
Command Line Code	Courier font	<code>java -jar EnterpriseDesigner.jar</code>
Hypertext Links	Blue text	For more information, see “Writing Conventions” on page 16 .
Notes	Bold Italic text	<i>Note: If a toolbar button is dimmed, you cannot use it with the selected component.</i>

1.5 Supporting Documents

The following SeeBeyond documents provide additional information about the eGate Integrator system as explained in this guide:

- *eGate Integrator Installation Guide*
- *eGate Integrator JMS Reference Guide*
- *eGate Integrator Release Notes*
- *eGate Integrator Tutorial*
- *SeeBeyond ICAN Suite Deployment Guide*
- *SeeBeyond ICAN Suite Primer*

For information on a specific add-on product (for example, an eWay Intelligent Adapter), see the User's Guide for that product. A complete list of eGate-related documentation is included in the *SeeBeyond ICAN Suite Primer*.

1.6 Online Documents

The documentation for the SeeBeyond ICAN Suite is distributed as a collection of online documents. These documents are viewable with the Acrobat Reader application from Adobe Systems. Acrobat Reader can be downloaded from:

<http://www.adobe.com>

1.7 The SeeBeyond Web Site

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

<http://www.seebeyond.com>

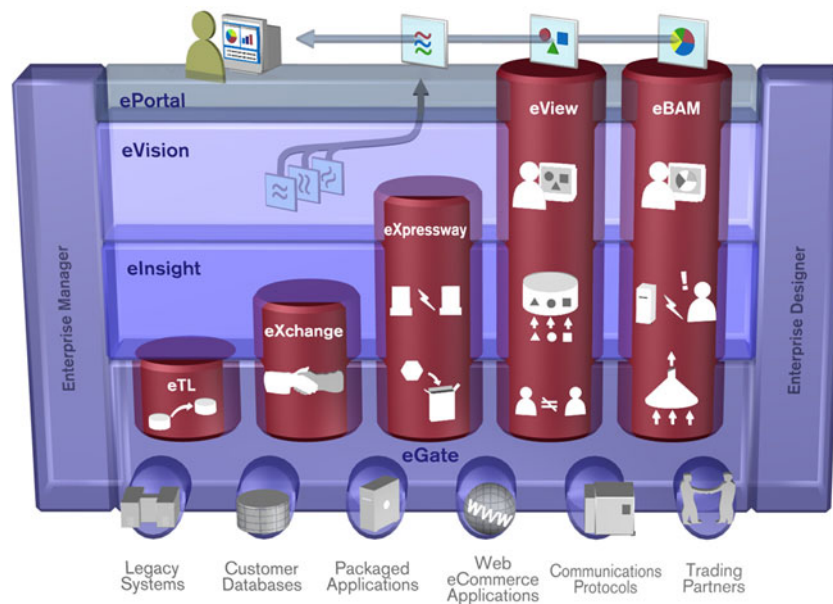
System Overview

This chapter provides an overview of the conceptual operation and general architecture of the eGate Integrator system.

2.1 Introduction

SeeBeyond's Integrated Composite Application Network (ICAN) Suite provides businesses with a comprehensive, unified eBusiness infrastructure to connect, integrate, and manage enterprise-wide software applications running on various computer systems. The full ICAN Suite is depicted in Figure 1.

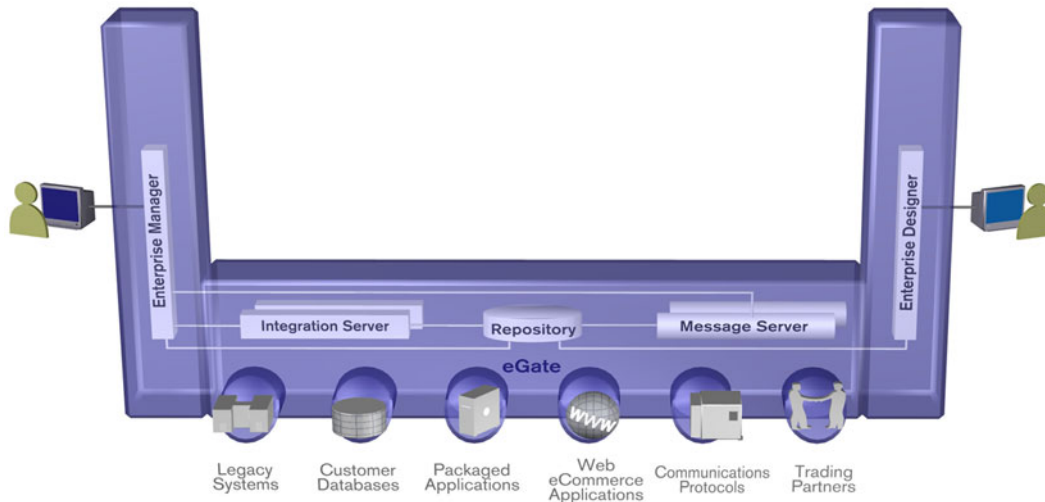
Figure 1 SeeBeyond ICAN Suite



SeeBeyond's eGate Integrator provides the "backbone" for the ICAN suite, integrating the various components of the Suite and all other connected components of the business enterprise. As shown in Figure 2, eGate Integrator includes the Enterprise Manager and Enterprise Designer, which provide graphical user interfaces for managing, configuring, and controlling the entire ICAN Suite and the business

processes running therein. See [Enterprise Manager](#) on page 25 and [Enterprise Designer](#) on page 23.

Figure 2 eGate Integrator



Other major constituents of eGate Integrator shown in Figure 2 are the Integration Server, the JMS IQ Manager, and the Repository, all of which will be described briefly later in this chapter. The flexibility of the eGate system allows the option of deploying it across a distributed network of hardware platforms, if desired, and running it on any combination of SeeBeyond, BEA WebLogic, and IBM WebSphere servers.

eGate Integrator can communicate with and link multiple applications and databases across a variety of different operating systems. eGate performs with a wide variety of hardware, message standards, operating systems, databases, and communication protocols in both real-time and batch (scheduled) integration modes.

2.2 Integration Model

SeeBeyond addresses application integration by means of an eGate Project, which contains the business logic required to solve the specific problem. The Project contains the various logical components and supporting information required to perform the routing, processing, and caching of messages containing the relevant data from one application to another. All Project information is stored in the Repository.

Projects are created using tools contained within the Enterprise Designer and, once deployed, can be run and monitored using Enterprise Manager. Projects can also be set up to be run from the business process level using the SeeBeyond eInsight Business Process Manager.

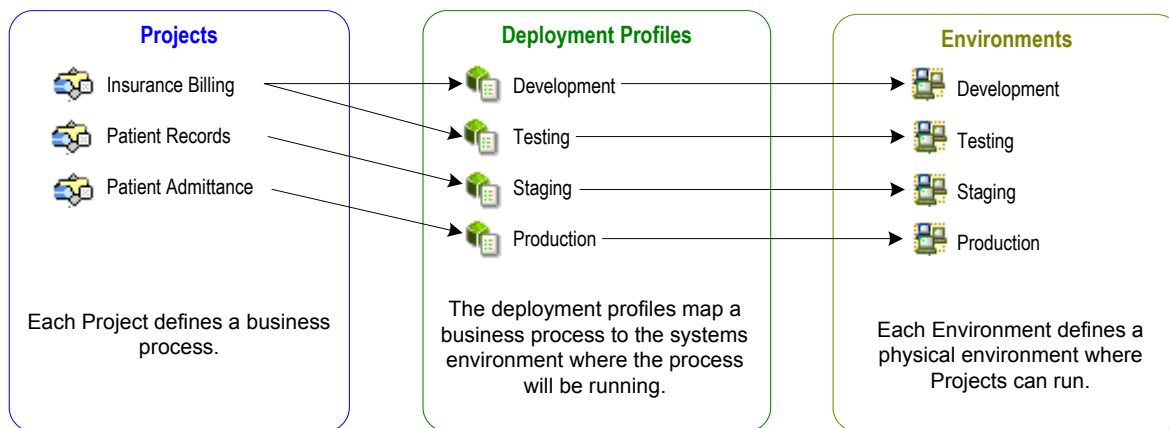
Projects are run within Logical Hosts, which are individual, runtime instances of eGate Integrator. Logical Hosts are defined within Environments, which represent the physical resources required to implement the Project. Projects are mapped to the individual Environments by means of Deployment Profiles, which are defined within

the Enterprise Designer and become part of the Project. Activating the Deployment Profile deploys the Project to the associated Environment.

This structure of Projects, Environments, and Deployment Profiles isolates each implementation into logical and physical components. This provides you with extensive flexibility and efficiency in designing eGate Integrator implementations. For example, once you build your Projects and Environments, you have the flexibility to change each component without having to make changes to the other component.

The finished Project, of course, will run in your production Environment; separate Environments, having the same structure as the production Environment, should be created for development and testing. You may also want some additional Environments, such as staging. The following figure illustrates the eGate Integrator implementation model using a healthcare-related example.

Figure 3 eGate Integrator Implementation Model



In the figure above, any of the Projects can be deployed to any of the Environments via the mapping defined in the deployment profiles. The example in the figure above shows that the patient admittance Project is already in the production phase and therefore was deployed using the production deployment profile. The patient records Project is in the staging phase and was therefore deployed to the staging Environment using the staging deployment profile. The insurance billing Project is still being developed and tested, and therefore it is deployed to development and testing via the development and testing profiles.

In broad outline, an eGate Integrator implementation includes the following steps:

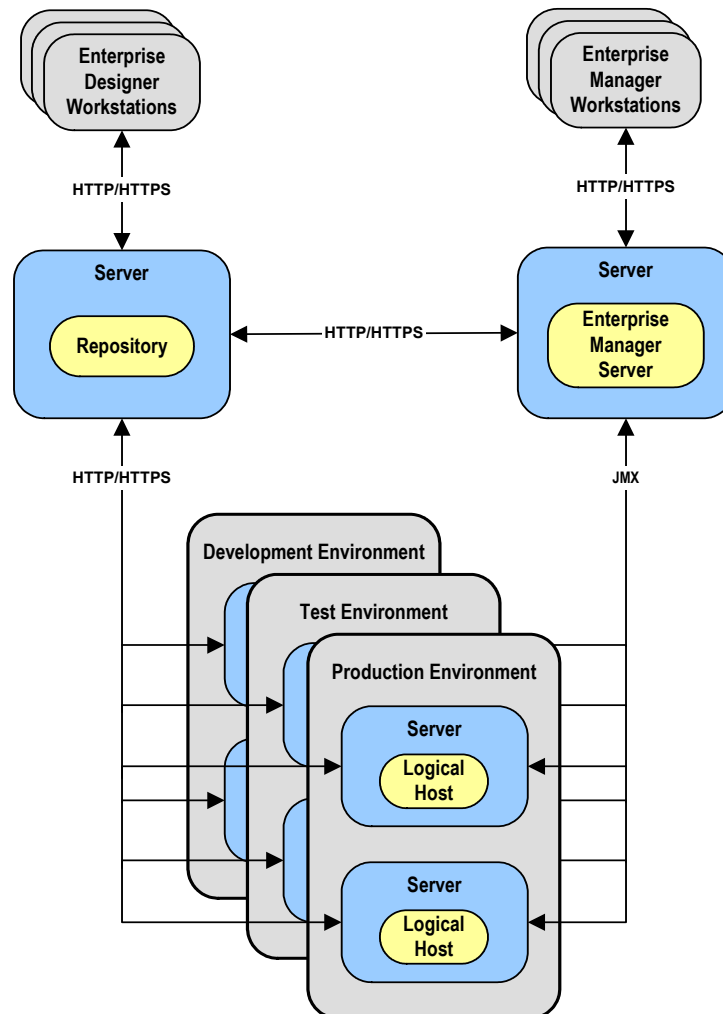
- 1 **Design your Project.**
- 2 **Define your Environments.**
- 3 **Create your Deployment Profiles.**
- 4 **Deploy the eGate Project.**

These implementation steps are all accomplished using the eGate Enterprise Designer, which is introduced in [Enterprise Designer](#) on page 23 and developed further in subsequent chapters.

2.3 System Architecture

eGate Integrator employs a flexible architecture that is ideal for distributed computing environments. As a result, the various components of an eGate Integrator system can reside on the same hardware platform (assuming adequate system resources), or be distributed across several different hardware platforms in the enterprise network. Figure 4 shows an example system implementation that is highly distributed.

Figure 4 Typical eGate Integrator System



2.3.1 Repository

The setup, components, and configuration information for the elements of a Project are stored in the Repository. The Repository also stores all of the product binary files that are required at runtime by the Logical Hosts. The components and configurations are downloaded to the Logical Host during the initial bootstrap process and as needed after design-time configuration changes are made.

As shown in Figure 4, a single Repository serves the entire enterprise. This common Repository is used for development, testing, and production purposes. Communication between the Repository and other eGate components can be configured to use either HTTP or HTTPS. The Enterprise Designer and Enterprise Manager clients can communicate with the Repository and Enterprise Manager servers through a firewall.

2.3.2 Environments

An eGate Environment represents the physical system required to implement a Project. It consists of a collection of Logical Hosts, capable of hosting components of the ICAN Suite, along with information about external systems involved in the implementation.

- **Logical Hosts**

Each Environment contains one or more Logical Hosts. A Logical Host contains one or more **integration servers**, which are the engines that run eGate Collaborations and eWays, and one or more **message servers**, which manage JMS topics (publish-and-subscribe messaging) and queues (point-to-point messaging).

- **External Systems**

An external system is a representation of a real, physical system that exists within the specific Environment, with configuration properties for locating and accessing that system.

In the example system shown in Figure 4, the production environment is split across two hardware platforms, each running a single Logical Host. Separate environments for development and testing should duplicate the structure of the production environment. The test environment should be supported by hardware similar to that supporting the production environment, to allow performance and load testing to give representative throughput results. The hardware supporting the development environment, however, does not usually have the same performance requirements as that supporting the test and production environments.

An eGate Project is created within the development environment, then migrated to the test environment, and finally to the production environment. This migration path is a necessary and highly critical practice in implementing a working system.

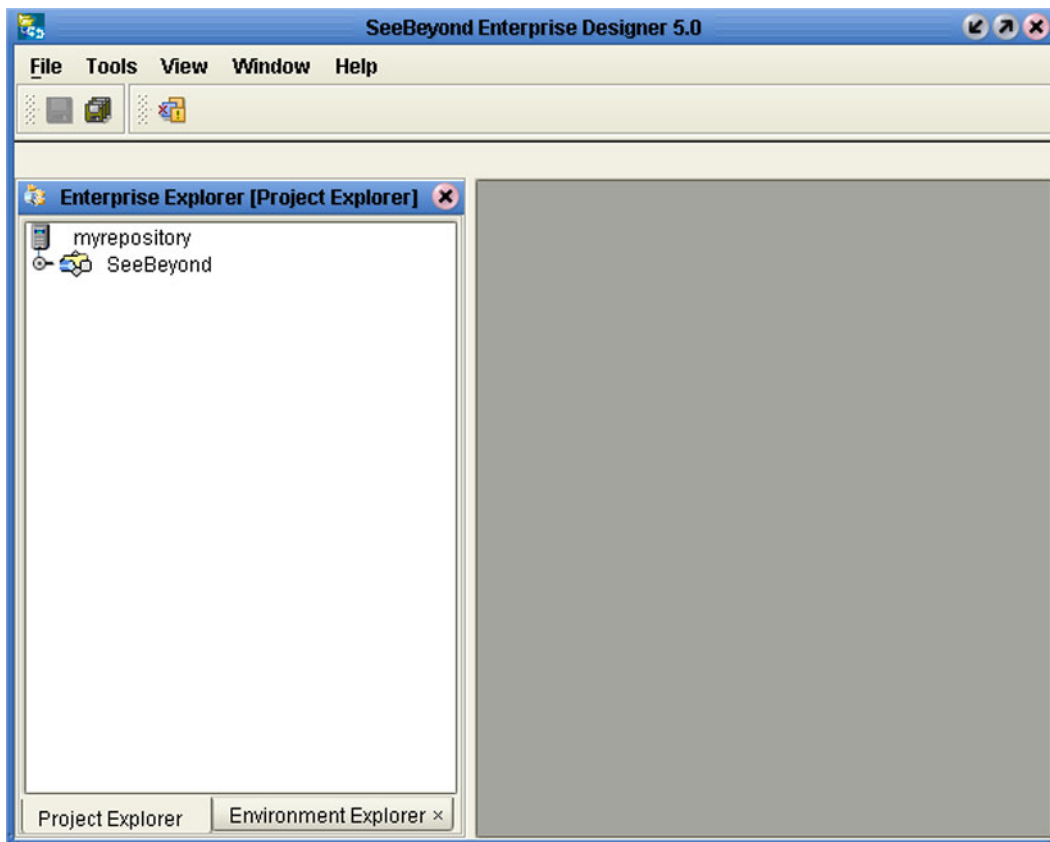
Note again that there is no requirement for the components shown in Figure 4 to run on separate systems; all could run on a single system, provided that resources (CPU, memory, and disk) are sufficient to support the concurrent usage.

2.4 User Interfaces

2.4.1 Enterprise Designer

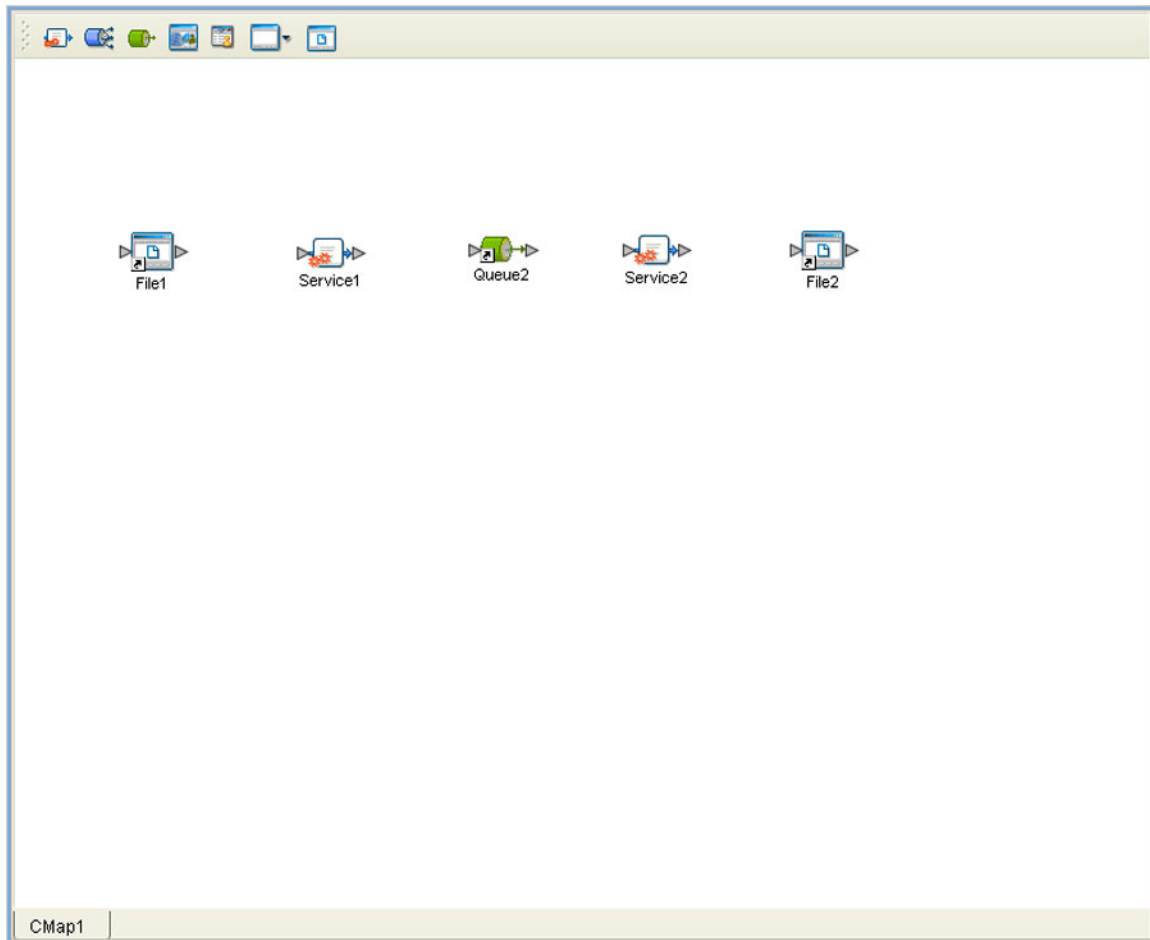
The Enterprise Designer graphical user interface (GUI) is used to create and configure the logical components and physical resources of an eGate Project. Through this GUI (see Figure 5), you can develop Projects to process and route data through an eGate Integrator system.

Figure 5 Enterprise Designer



The major features of the Enterprise Designer are the Enterprise Explorer on the left, and an editor panel on the right—which is initially blank. The Enterprise Explorer follows the familiar Windows Explorer format, displaying a tree structure. The editor panel displays a variety of editors, depending upon what component is selected in the Enterprise Explorer. The Connectivity Map Editor (see Figure 6) provides a graphic example of one of these, in which logical components of a Project are created and connected.

Figure 6 Connectivity Map Editor



The features and usage of the Connectivity Map Editor are described in [Creating Projects](#) on page 47. Other editors are displayed for creating and modifying Object Type Definitions, Collaboration Definitions, Deployment Profiles, and other Project components.

The Enterprise Designer also includes the design-time functionality for other ICAN products, such as eXchange. For more information on using other ICAN products in the Enterprise Designer, see the product documentation for those products.

For more information on the Enterprise Designer, see [Enterprise Designer](#) on page 26 .

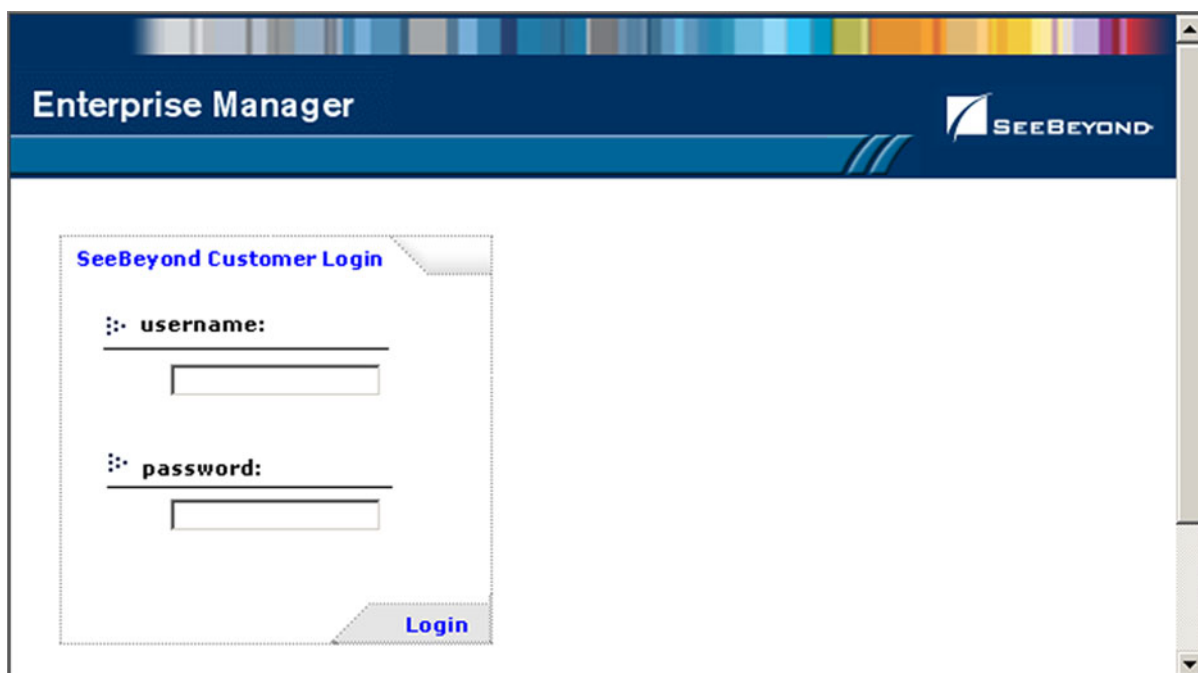
2.4.2 Enterprise Manager

The Enterprise Manager is a Web-based application you use for:

- Managing and monitoring eGate runtime components.
- Installing ICAN Suite products into the Repository.
- Downloading and installing products from the Repository.
- Accessing other Web-based ICAN Suite products.
- Accessing ICAN Suite product documentation.

The Enterprise Manager (see Figure 7) is accessed via Microsoft Internet Explorer.

Figure 7 SeeBeyond Enterprise Manager Login



The screenshot shows a web browser window displaying the SeeBeyond Enterprise Manager login page. The page features a dark blue header with the text "Enterprise Manager" on the left and the "SEE BEYOND" logo on the right. Below the header is a white box titled "SeeBeyond Customer Login". Inside this box, there are two input fields: "username:" and "password:", each with a small icon to its left. Below the password field is a "Login" button.

For more information on the Enterprise Manager, see [Enterprise Manager](#) on page 146.

For more information on Web Services capability, see [Web Services](#) on page 63.

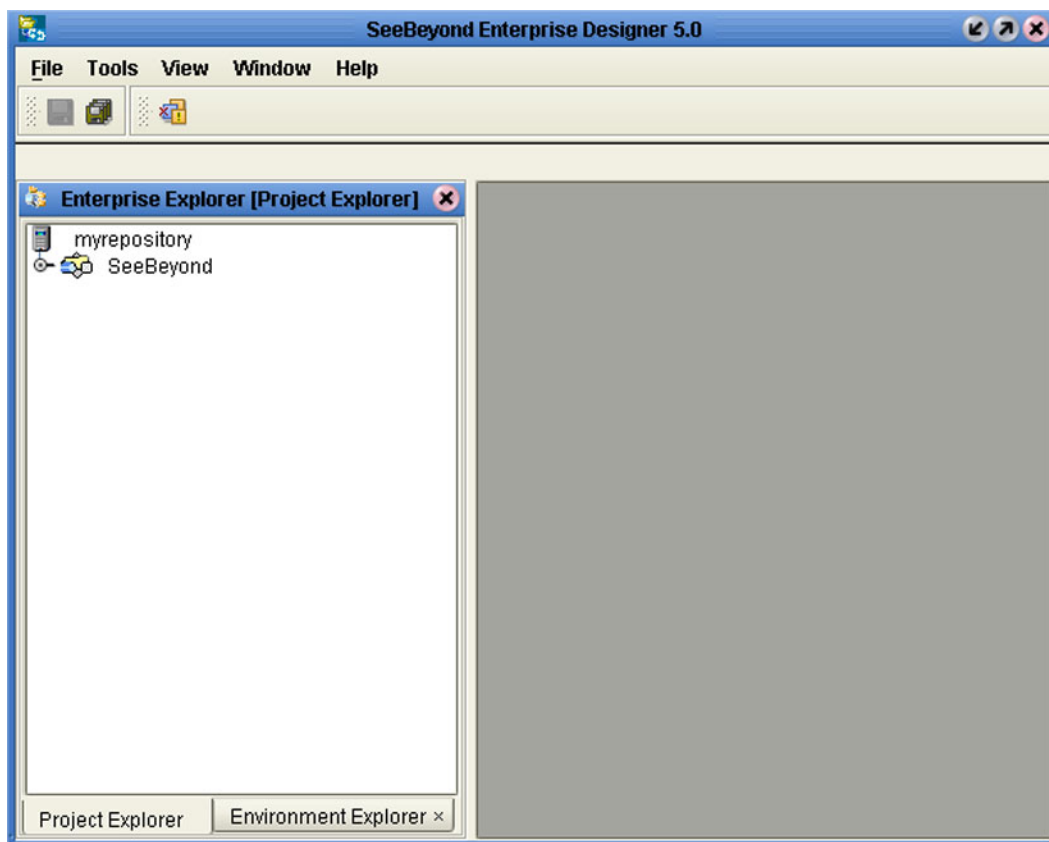
Enterprise Designer

This chapter describes the various features of the Enterprise Designer.

3.1 Overview

The Enterprise Designer graphical user interface (GUI) is used to create and configure the logical components and physical resources of an eGate Project. Through this GUI (see Figure 8), you can develop Projects to process and route data through an eGate Integrator system.

Figure 8 SeeBeyond Enterprise Designer



The major features of the Enterprise Designer are the Enterprise Explorer on the left, and an editor panel on the right—which is initially blank. The Enterprise Explorer follows the familiar Windows Explorer format, displaying a tree structure. The Enterprise Explorer provides two views of the ICAN system, described in the following sections of this chapter:

- **Project Explorer** on page 31
- **Environment Explorer** on page 36

The editor panel displays a variety of editors, depending upon what component is selected in the Enterprise Explorer. These editors are described in the following sections of this chapter:

- **Connectivity Map Editor** on page 43
- **OTD Editor** on page 44
- **Environment Editor** on page 45
- **Deployment Editor** on page 46

The Enterprise Designer also contains the customary graphical interface features, including the following:

- **Menu Bar** on page 28 describes the options contained in the individual menus.
- **Toolbar** on page 30 describes the functionality of the toolbar icons.
- **Browser Buttons** on page 30 describes the browser buttons that appear throughout the Enterprise Designer, in various wizards and dialog boxes.

The procedure for invoking the Enterprise Designer is described in **Starting Enterprise Designer** on page 49.

3.2 Menu Bar

The menu bar provides access to a variety of options for managing your Project. The individual menus are described in the following tables.

3.2.1 File Menu

Table 2 File Menu Options

Option	Function
Restore	Displays a dialog box with which you can locate and select a Repository archive file to restore in the Enterprise Designer. See Restoring a Repository on page 124.
Backup	Displays a dialog box with which you can select a location on your computer to save a copy of the Repository archive. See Backing Up a Repository on page 124.
Save	Saves changes to the selected Project.
Save All	Saves changes to all Projects.
Exit	Closes the Enterprise Designer.

3.2.2 Tools Menu

Table 3 Tools Menu Options

Option	Function
Impact Analysis	Displays a dialog box with which you can view how one component of a Project impacts other components. See Impact Analysis on page 118.
Update Center	Displays a dialog box with which you can check for program updates. See the <i>eGate Integrator Installation Guide</i> .

3.2.3 View Menu

Table 4 View Menu Options

Option	Function
Environment Explorer	Activates the Environment Explorer tab on the Enterprise Explorer. See Environment Explorer on page 36.
Project Explorer	Activates the Project Explorer tab on the Enterprise Explorer. See Project Explorer on page 31.

3.2.4 Window Menu

Table 5 Window Menu Options

Option	Function
Cascade	Displays all open windows so that each window slightly overlaps the others in the Project Editor.
Tile	Displays all open windows in a stacked tile pattern.
Horizontal Layout	Displays all open windows from top to bottom.
Vertical Layout	Displays all open windows from left to right.
Minimize All	Minimizes all open windows so that only the title bar displays at the bottom of the Project Editor.
Restore All	Returns minimized windows to their original position on the Project Editor.
Close All	Closes all open windows.




3.2.5 Help Menu

Table 6 Help Menu Options

Option	Function
Contents	Displays the online help for eGate Integrator.
Help Set	Displays additional online help options.

3.3 Toolbar






Table 7 Enterprise Designer Toolbar Icons

Icon	Function
	Save saves changes to the selected Project (inactive if no changes have been made).
	Save All saves changes to all Projects (inactive if no changes have been made).
	Displays the Impact Analyzer dialog box, which allows you to view how one component of a Project impacts other components.

3.4 Browser Buttons

The following buttons are used throughout the Enterprise Designer, in wizards and file selection dialog boxes. They correspond to standard Windows browser buttons.

Table 8 Browser Buttons

Button	Function
	Up One Level returns you to the parent folder or directory.
	Home returns you to the root folder or directory.
	Create New Folder creates a new folder under the current folder.
	List displays folder/file names only.
	Details displays details of the folders or files (name, type, date last modified, etc.).

3.5 Enterprise Explorer

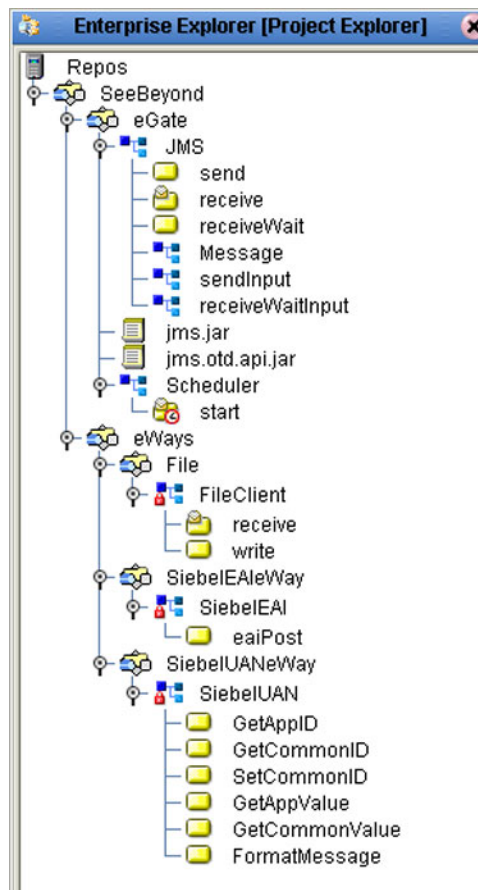
The Enterprise Explorer organizes the components of a Project into tabs that display different views of an eGate system.

- **Project Explorer** on page 31 deals with logical components.
- **Environment Explorer** on page 36 deals with physical resources, including the Logical Host and Integration Server.

3.5.1 Project Explorer

A Project consists of logical constructs and configurations designed to solve some or all of a business problem. The **Project Explorer** tab includes folders and icons that represent the names and contents of Projects. Some example components of a Project are shown in Figure 9.

Figure 9 Enterprise Explorer: Project Explorer View









Salient features of the Project Explorer are:

- **Project Explorer Icons** on page 32
- **Context Menus** on page 33

3.5.2 Project Explorer Icons

The icons described in Table 9 appear in the Project Explorer.

Table 9 Project Icons

Icon	Description
	Represents the Repository , which is the database where all Projects and contents are saved.
	Represents the Project or subproject.
	Represents a Connectivity Map , which contains the business logic and information about the data transmission. .
	Represents a Project variable or constant .
	Represents an Object Type Definition (OTD) file. A lock displayed in the lower-left corner indicates that the OTD is currently checked into the version control system.
	Represents a Deployment Profile , which specifies how Project components will be deployed in an Environment.

3.5.3 Context Menus

Right-clicking on a component in the Project Explorer displays a context menu for that component. Included here are descriptions of options for the following component context menus:

- [Repository Menu](#) on page 33
- [Project Menu](#) on page 34
- [Connectivity Map Menu](#) on page 35

Repository Menu

Figure 10 Repository Menu

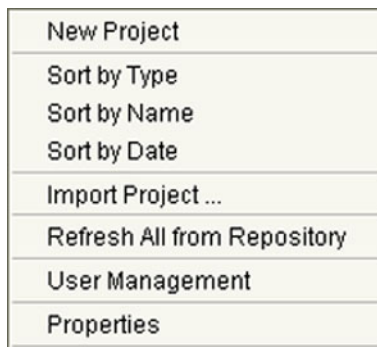


Table 10 Repository Menu Options

Option	Function
Project	Adds a new Project to the Project Explorer tab.
Sort by Type	Places Project components in order by grouping component types.
Sort by Name	Places Projects and Project components in alphabetical order.
Sort by Date	Places Projects in order by creation date from oldest Project to newest.
Import Project	Displays a dialog box, which you can use to locate and select a Repository archive file to import into the Enterprise Designer.
Refresh All from Repository	Updates Enterprise Designer with Project/Environment configurations stored in the Repository. (Open editors are not refreshed.)
User Management	Displays the User Management dialog box, where you can manage user access to the Repository with options for adding, modifying, and deleting users. See Configuration User Management on page 131.
Properties	Displays a dialog box (the exact name of the dialog box mirrors the name of your Repository). Use this dialog box to manage Enterprise Designer users and their access to the Repository. See Configuration User Management on page 131.

Project Menu

Figure 11 Project Menu

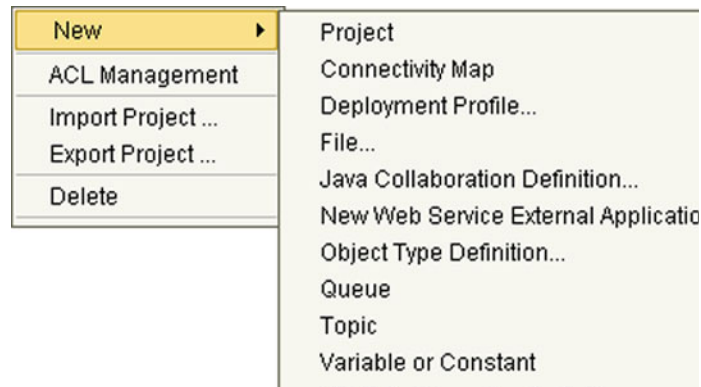


Table 11 Project Menu Options

Option	Option	Function
New	Project	Adds a Subproject folder to the selected Project.
	Connectivity Map	Adds a Connectivity Map to the Project. See Using the Connectivity Map Editor on page 52.
	Deployment Profile	Displays a dialog box with which you can assign a Deployment Profile to the selected Project. See Using the Deployment Editor on page 92.
	File	Displays a dialog box with which you can create an external file to use with the Project.
	New Web Service ...	Adds a third-party Web service application to the Project Explorer. See Example Web Services Project on page 65.
	Object Type Definition	Displays the OTD Wizard , with which you can create an Object Type Definition (OTD) file. See Using the OTD Wizard on page 71 for more information.
	Queue	Adds a queue to your Project.
	Topic	Adds a topic to your Project.
	Variable or Constant	Displays a dialog box with which you can add a constant or variable icon to your Project.
ACL Management		Displays the ACL Properties dialog box, with which you can assign read and/or write privileges to users for the selected Project. See ACL Management on page 136.
Import Project		Displays a dialog box with which you can locate and select a Project file to import into the Enterprise Designer.
Export Project		Displays a dialog box with which you can select a location on your computer to save a copy of the Project.
Delete		Deletes the selected Project.

Connectivity Map Menu

Figure 12 Connectivity Map Menu

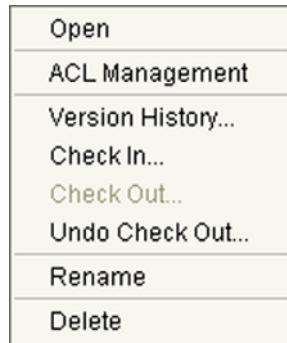


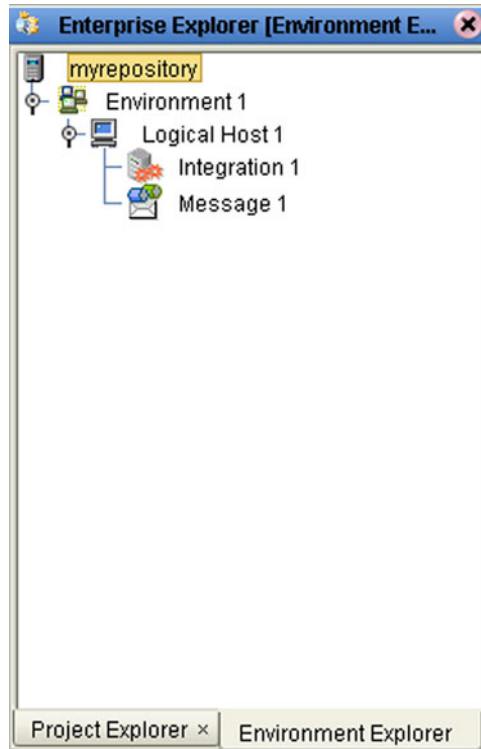
Table 12 Connectivity Map Menu Options

Command	Function
Open	Opens the Connectivity Map Editor for the selected Connectivity Map. See Using the Connectivity Map Editor on page 52.
ACL Management	Displays the ACL Properties dialog box, with which you can assign read and/or write privileges to users for the selected Project. See ACL Management on page 136.
Version History	Displays a dialog box with which you can track the version history for OTDs and Collaboration Definitions. Version control allows users to maintain multiple versions of the same OTD and Collaboration Definition files. See Version Control on page 121 for more information.
Check In	Displays a dialog box, with which you can check in the current version of a Project. Refer to Checking In a Project or Environment Component on page 122 for more details.
Check Out	Displays a dialog box with which you can check out a version of a Project. See Checking Out a Project or Environment Component on page 123 for more information.
Undo Check Out	Reverses the Check Out command, returns you to the previous state.
Rename	Allows you to rename the selected Connectivity Map.
Delete	Displays a dialog box in which you confirm that you want to delete the selected Connectivity Map. Clicking Yes then deletes the Connectivity Map.

3.5.4 Environment Explorer

An Environment consists of Logical Hosts capable of hosting eGate components and information about external systems which may be involved with an eGate configuration.

Figure 13 Enterprise Explorer: Environment Explorer View











Salient features of the Project Explorer are:

- [Environment Explorer Icons](#) on page 37
- [Context Menus](#) on page 38

3.5.5 Environment Explorer Icons

The icons described in Table 13 appear in the Environment Explorer.

Table 13 Environment Icons

Icon	Function
	Represents the Repository , which is the database where all Projects and contents are saved.
	Represents the Environment , which contains Logical Hosts and information about external systems.
	Represents a Logical Host , which contains the software and other installed components that are required at runtime.
	Represents an external file .
	Represents an Environmental constant, which you can use to automate eWay and message destination configuration changes.
	Represents a Scheduler component of an Environment, which you can use to set data transfer to occur at set intervals.
	Represents an Integration Server .
	Represents a JMS IQ Manager or third-party message server , which is used to store and forward eGate system messages.

3.5.6 Context Menus

Right-clicking on a component in the Environment Explorer displays a context menu for that component. Included here are descriptions of options for the following component context menus:

- [Repository Menu](#) on page 38
- [Environment Menu](#) on page 39
- [Logical Host Menu](#) on page 40

Repository Menu

Figure 14 Repository Menu



Table 14 Repository Menu Options

Option	Function
New Environment	Displays a dialog box with which you can create a new Environment.
Save Changes to Repository	Saves all changes made in the Environment Explorer.
Refresh All from Repository	Updates Enterprise Designer with Project/Environment configurations stored in the Repository. (Open editors are not refreshed.)

Environment Menu

Figure 15 Environment Menu

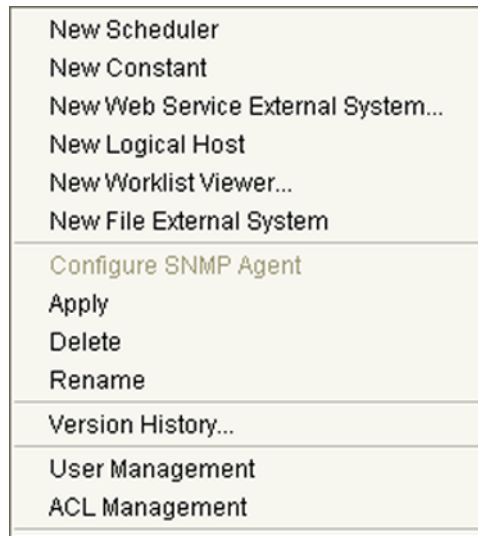


Table 15 Environment Menu Options

Option	Function
New Scheduler	Displays a dialog box with which you can add a new scheduling component to the selected Environment.
New Constant	Displays a dialog box with which you can add a constant to the Environment.
New Web Service ...	Adds a third-party Web service application to the Project Explorer. See Example Web Services Project on page 65.
New Logical Host	Adds a new Logical Host to the selected Environment.
New Worklist Viewer	See the <i>eInsight Enterprise Service Bus User's Guide</i> for information.
New File External System	Displays a dialog box with which you can add a new external file to the selected Environment.
Configure SNMP Agent	Displays a dialog box with which you can modify agent properties.
Apply	Saves changes to the selected Environment.
Delete	Displays a dialog box in which you confirm that you want to delete the selected Environment. Clicking Yes then deletes the Environment.
Rename	Allows you to rename the selected Environment.
Version History	Displays a dialog box with which you can track the version history for Environments. Version control allows users to maintain multiple versions of the same Environment. See Version Control on page 121 for more information.

Table 15 Environment Menu Options

Option	Function
User Management	Displays a dialog box with which you can manage Repository access. See Configuration User Management on page 131.
ACL Management	Displays the ACL Properties dialog box, with which you can assign read and/or write privileges to users for the selected Project. See ACL Management on page 136.

Logical Host Menu

Figure 16 Logical Host Menu

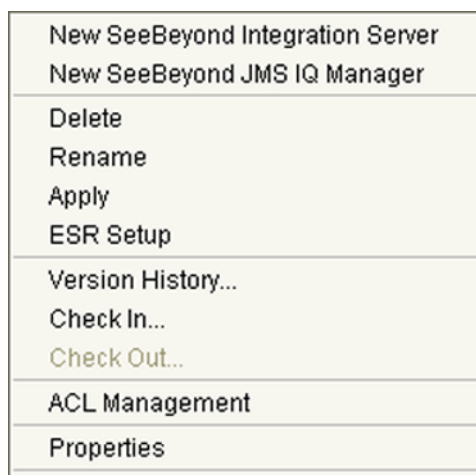


Table 16 Logical Host Menu Options

Option	Function
New SeeBeyond Integration Server	Adds a new SeeBeyond integration server to the selected Logical Host.
New SeeBeyond JMS IQ Manager	Adds a new SeeBeyond JMS IQ Manager to the selected Logical Host.
Delete	Displays a dialog box in which you confirm that you want to delete the selected Logical Host. Clicking Yes then deletes the Logical Host.
Rename	Allows you to rename the selected Logical Host.
Apply	Saves changes to the selected Logical Host.
ESR Setup	Displays a dialog box with which you can select emergency software releases (ESRs) to add to the Environment.
Version History	Displays a dialog box with which you can track the version history for Logical Hosts. Version control allows users to maintain multiple versions of the same Logical Host. See Version Control on page 121 for more information.

Table 16 Logical Host Menu Options

Option	Function
Check In	Displays a dialog box, with which you can check in the current version of an Environment. Refer to Checking In a Project or Environment Component on page 122 for more details.
Check Out	Displays a dialog box with which you can check out a version of an Environment. See Checking Out a Project or Environment Component on page 123 for more information.
ACL Management	Displays the ACL Properties dialog box, with which you can assign read and/or write privileges to users for the selected Environment See ACL Management on page 136.
Properties	Displays a dialog box with which you can modify the default settings for the selected Logical Host.

Note: If you are using BEA WebLogic and/or IBM WebSphere, the Integration Servers and JMS Message Servers for these products will also appear in the context menu.

3.6 Enterprise Designer Editors

This part of the Enterprise Designer is empty when you start a new Project. As you proceed through the development process, different editors are displayed in this area. These are described briefly as follows:

- **Connectivity Map Editor** on page 43
- **OTD Editor** on page 44
- **Environment Editor** on page 45
- **Deployment Editor** on page 46

Note: See the *eGate Tutorial* for detailed information and instructions for setting up a Project.

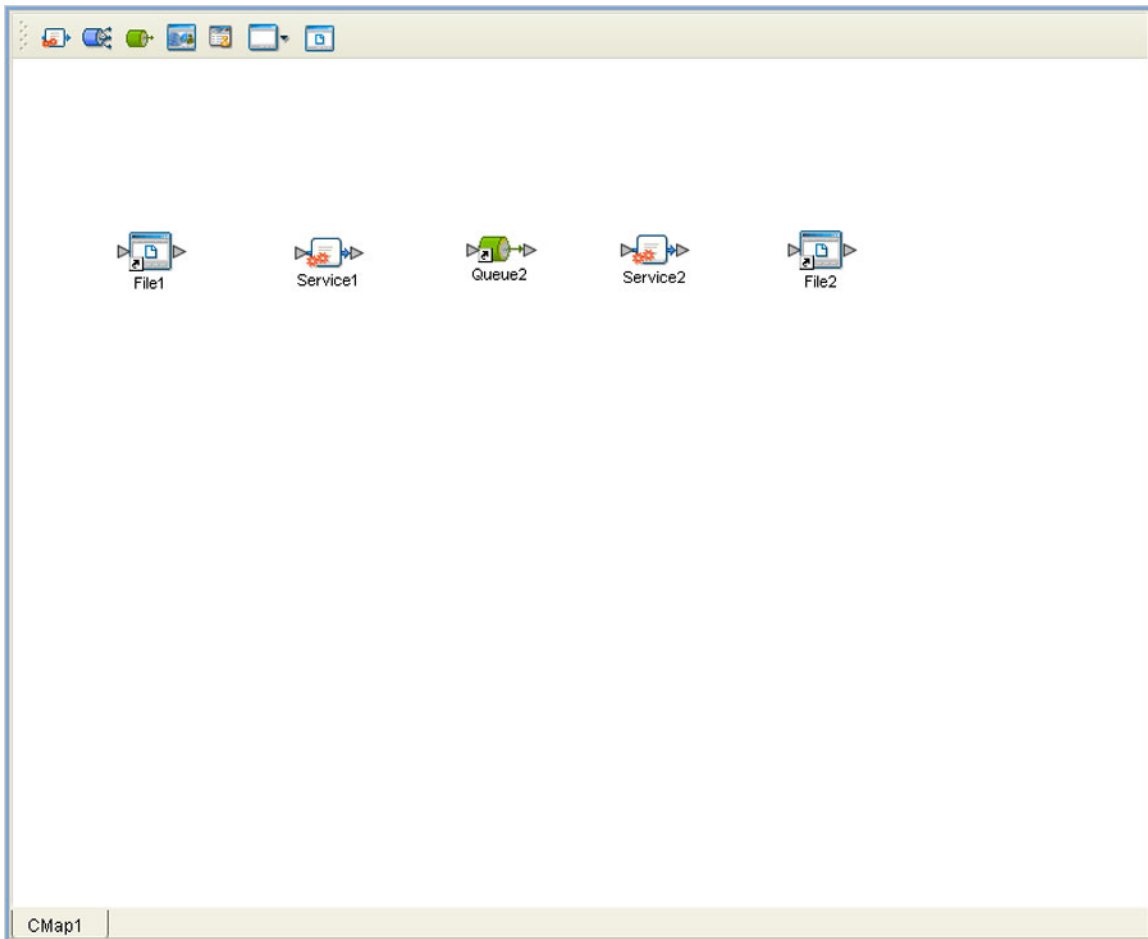
3.6.1 Connectivity Map Editor

A Connectivity Map is a graphical representation of your Project, containing the various logical components comprising the Project and the links between them. The Connectivity Map Editor, shown in Figure 17, allows you to create your Project by simply dragging and dropping icons onto a Project canvas and then connecting them to form data paths. You then can configure the components by means of dialog boxes that are displayed by clicking on the component icons.

Note: *It is best to create your Collaboration Definitions before using the Connectivity Map to connect components.*

See [Using the Connectivity Map Editor](#) on page 52 for detailed information.

Figure 17 Connectivity Map Editor

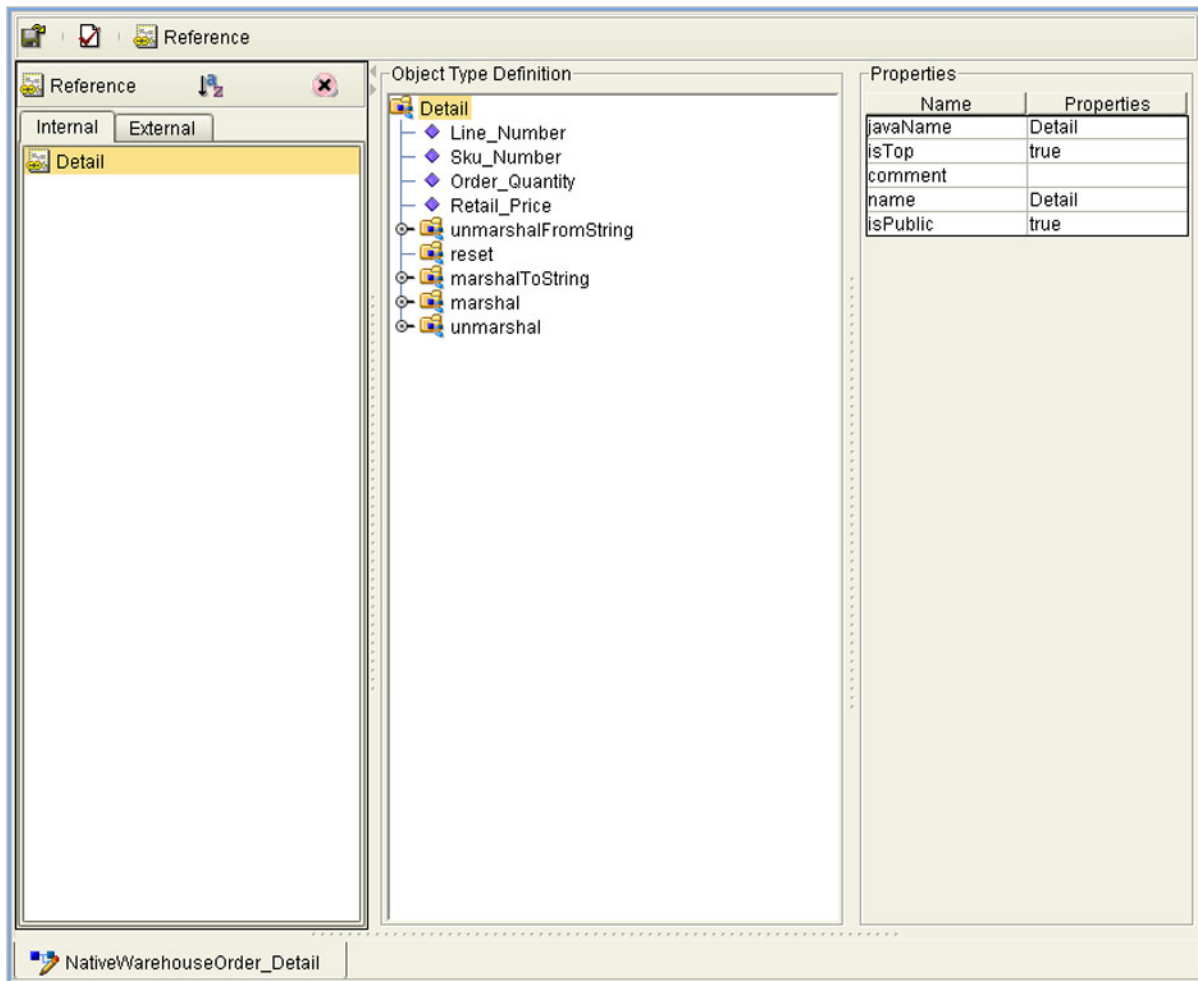


3.6.2 OTD Editor

The OTD Editor window, as shown in Figure 18, displays the source files used to create the Object Type Definitions (OTDs) to use with a Project. You use an OTD wizard tool to compile OTD files and add them to the **Project Explorer** tab.

See [Using the OTD Editor](#) on page 108 for detailed information.

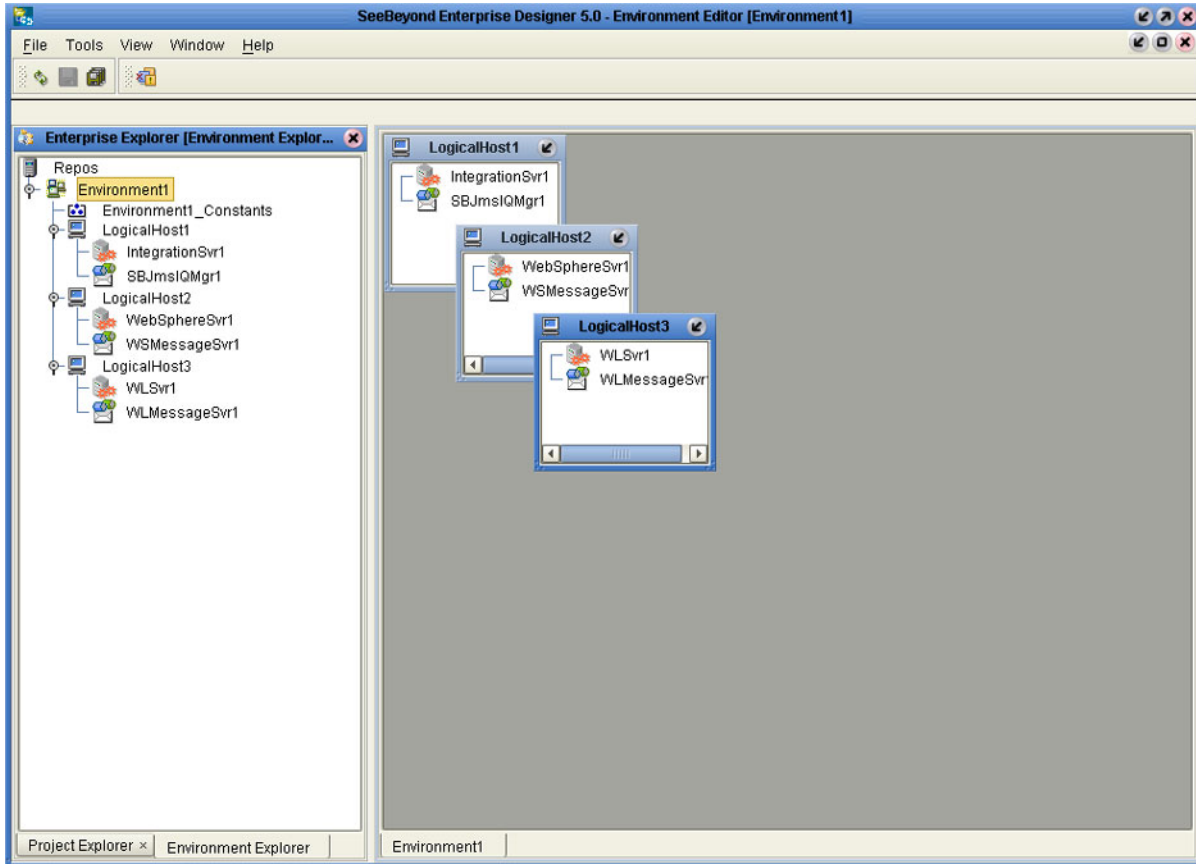
Figure 18 OTD Editor



3.6.3 Environment Editor

Clicking on an Environment icon in the Environment Explorer invokes the Environment Editor, which provides a canvas in which you can create and customize an Environment. Here you can see the various components (Logical Hosts, servers, and external systems) included in the selected Environment. An environment containing example Logical Hosts is shown in Figure 19.

Figure 19 Environment Editor

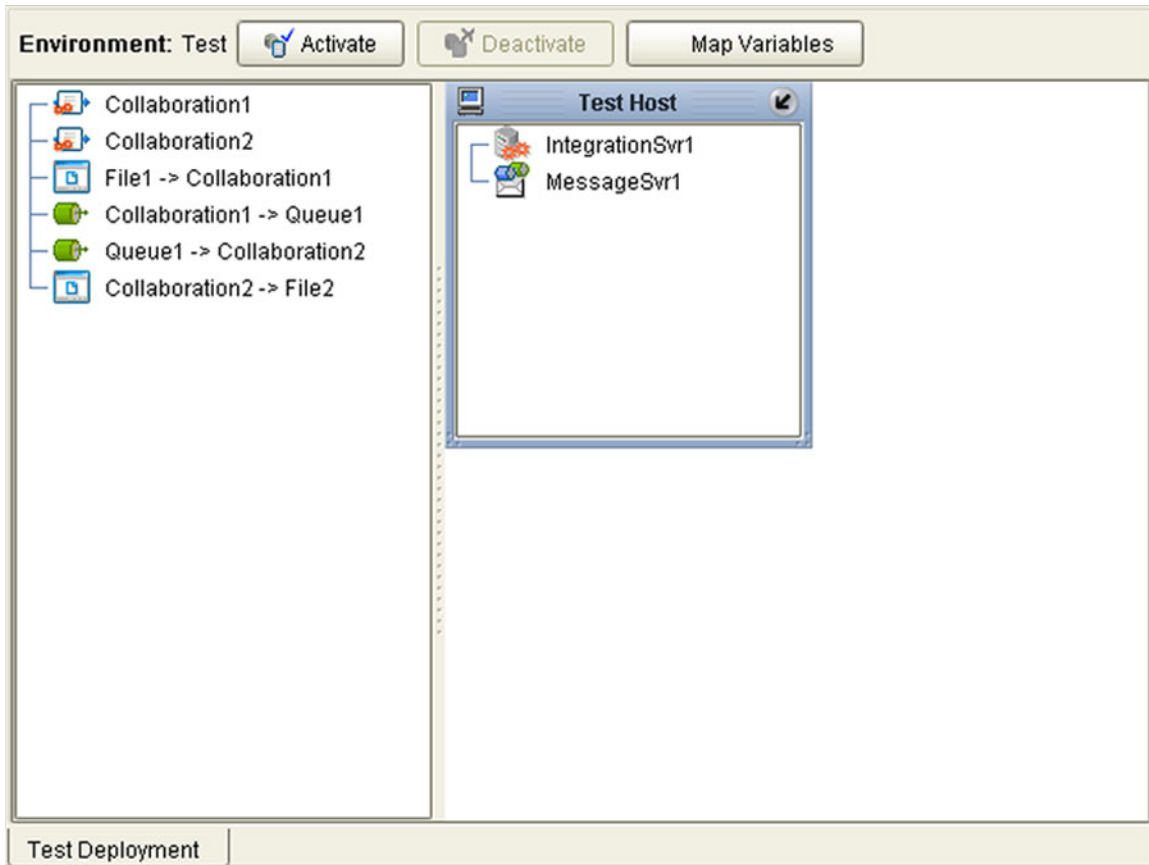


Note: Unlike changes to Project-related configuration properties, changes to Environment-related properties do not require redeployment.

3.6.4 Deployment Editor

The Deployment Editor, as shown in Figure 20, contains information about how Project components will be deployed in an Environment. See [Using the Deployment Editor](#) on page 92 for detailed information

Figure 20 Deployment Editor



Creating Projects

This chapter describes the use of the Enterprise Designer in defining your Project.

4.1 Overview

An eGate Project represents the logical system designed to solve either all or part of a business problem. Projects are created using tools contained within the Enterprise Designer (see [Enterprise Designer](#) on page 26). They are deployed to specific Logical Hosts in specific Environments by means of Deployment Profiles (see [Environments and Project Deployment](#) on page 88). An end-to-end example of Project implementation is given in the *eGate Integrator Tutorial*.

A typical Project contains:

- Object Type Definitions (OTDs), that define object encoding rules.
- Collaborations, based on Collaboration Definitions, that process data.
- Message destinations, including topics and queues, that store data.
- Connections, including eWays and JMS Clients, that specify logical connections.
- Proxies for external applications.
- Connectivity Maps, that display the business logic.
- Deployment Profiles, that map the Project to specific Environments.

An eGate Project is related to an Activity in an eInsight business process. Components developed for use in one Project can be used in another, and a Project can internally reference another Project.

4.2 Project Components

As previously mentioned, an eGate Project represents the logical system designed to solve either all or part of a business problem. Projects are created using tools contained within the Enterprise Designer such as the Connectivity Map Editor. Various components of a Project shown in the figure include:

- **External Application Proxies**

The basic purpose of eGate Integrator is to facilitate the interchange of data between external business applications. These business applications are collectively referred to as External Applications, and are represented in the Project by logical proxies for the specific applications involved. An External Application can be identified with an ERP application such as SAP or PeopleSoft, a DBMS such as Oracle or SQL, or with a particular communications protocol, such as TCP/IP or HTTPS.

- **Collaborations**

A logical operation taking place between a Message Destination and an External Application, two Message Destinations, or two External Applications, having a publication and subscription relationship with those entities. The operation is defined by a Collaboration Definition.

- **Message Destinations**

A Message Destination is a container for stored data, and can follow either the topic or queue JMS model. A **topic** is a message destination that conforms to the publish-and-subscribe messaging paradigm. A **queue** is a message destination that conforms to the point-to-point messaging paradigm.

- **Logical Connections**

An **eWay Intelligent Adapter** provides the logical connection between external systems (applications or protocols) and a service. The identity of the External Application placed in a Connectivity Map dictates the type of eWay adapter assigned to the link between it and a Service such as a Collaboration, and can be accessible via multiple transport mechanisms.

A **JMS Client Connection** specifies the connection properties for the linked service (for example, publisher or subscriber).

Behind the scenes, and not explicitly shown in a Connectivity Map, are other Project components, such as:

- **Collaboration Definitions**

A Collaboration Definition defines the logical operation taking place in the related Collaboration. It is based on an Object Type Definition.

- **Object Type Definitions**

Object Type Definitions (OTDs) are sets of rules that define the encoding of an object. They describe messages that are propagated through eGate, and the methods available for operating on them, and also interactions with external APIs.

4.3 Starting Enterprise Designer

To start the Enterprise Designer

- 1 Start the Enterprise Designer using one of these two methods:
 - If you are connecting to a Repository on an HP NonStop Server, open a command prompt and change directories to `C:\ICAN50\edesigner\bin`, and type the following command:

```
runed hostname port rep_name
```

where:

hostname is the TCP/IP host name of the server where you installed the Repository—not the name of the Repository itself,

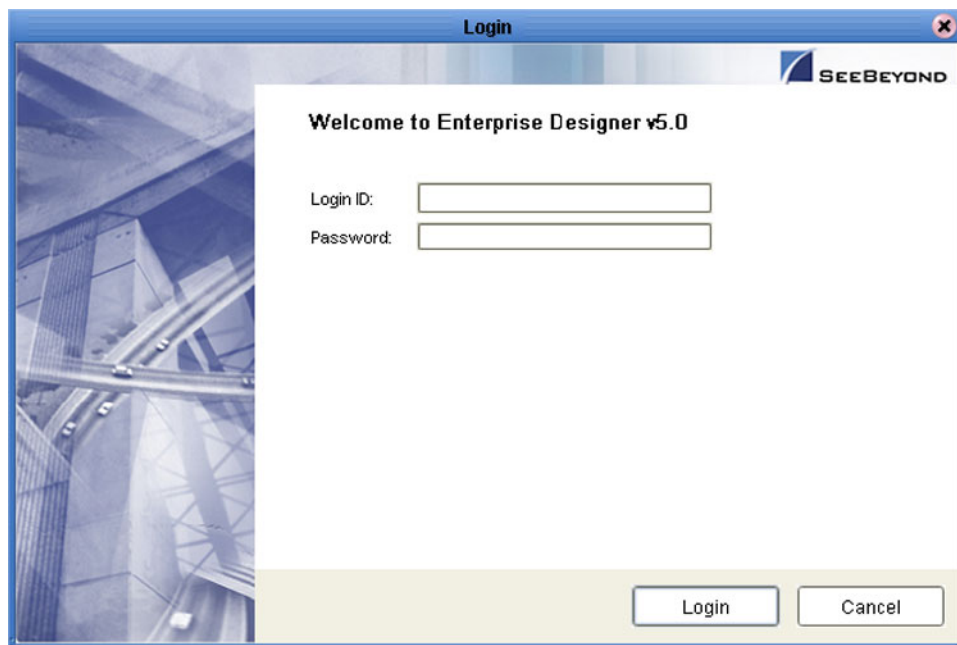
port is the port number of the Repository, and

rep_name is the name of the Repository.

This displays the dialog box shown in Figure 21.

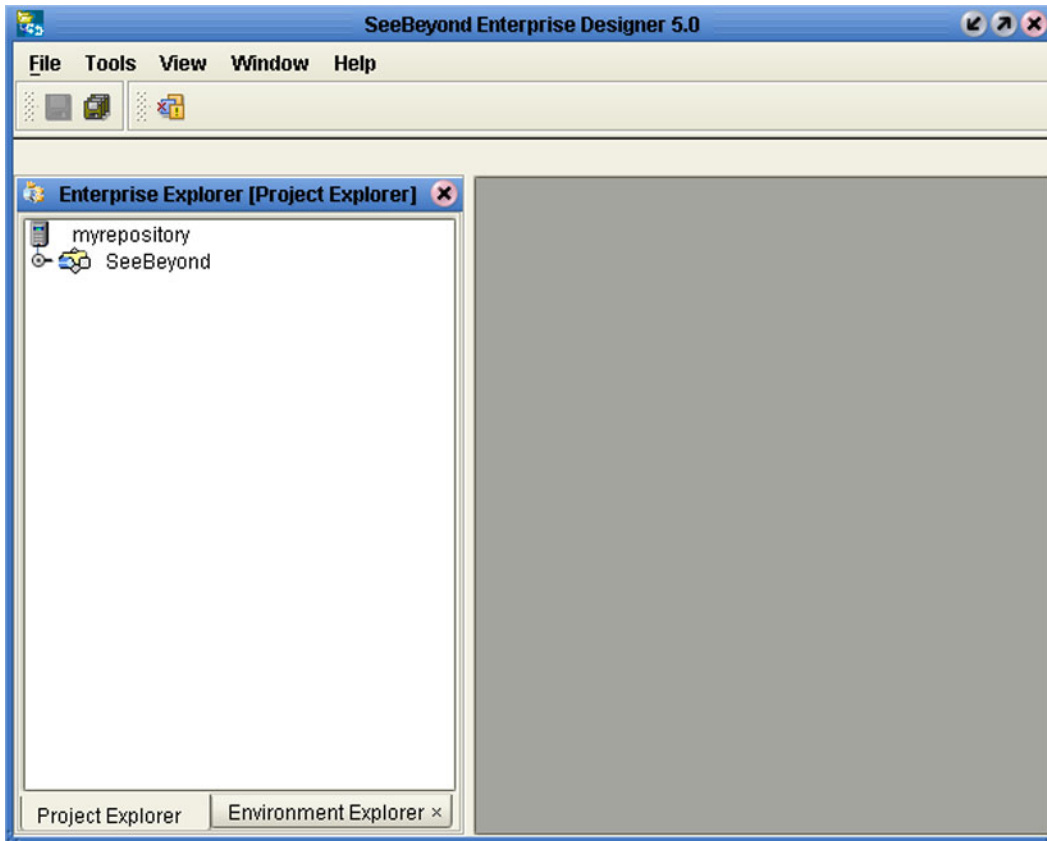
- If you are connecting to a Repository on any other platform, run the batch file `C:\ICAN50\edesigner\bin\runed.bat` to display the *Login* dialog box shown in Figure 21.

Figure 21 Login Dialog Box



- 2 Click in the *Login ID* text box, and enter your login ID.
- 3 Tab to the *Password* text box, and enter your password.
- 4 Click **Login** to complete the login process and display the Enterprise Designer GUI shown in Figure 22.

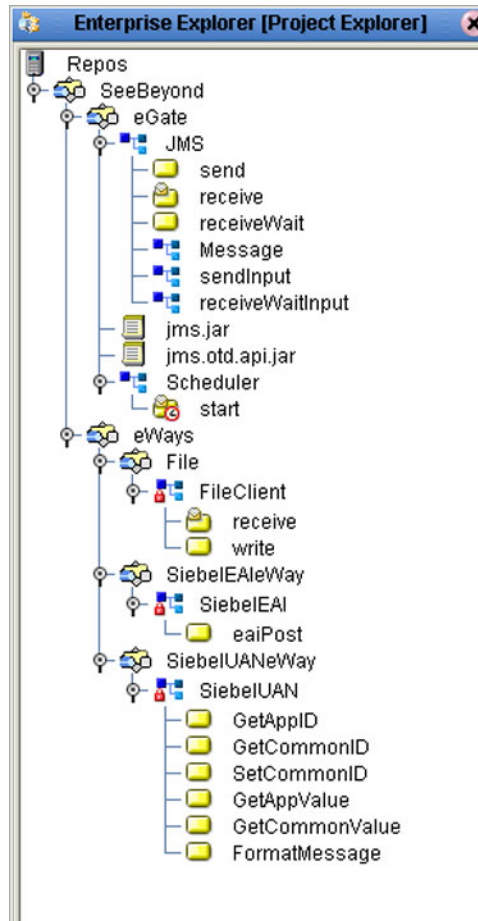
Figure 22 SeeBeyond Enterprise Designer



The editor panel—which is initially blank—displays a variety of editors, depending upon what component is selected in the Enterprise Explorer. The use of these editors is described in the following sections of this chapter.

4.4 Using the Project Explorer

Figure 23 Project Explorer

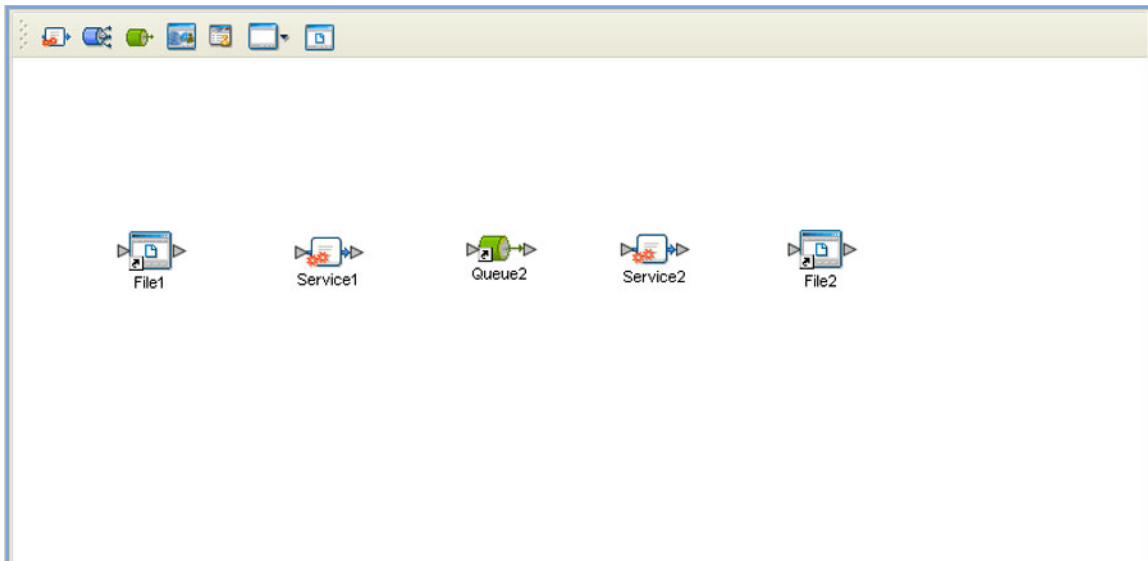


Note: Any changes made to the names of the Collaborations should be done when the Collaboration is created. If the name is changed later, the Collaboration should be opened, regenerated if applicable, and saved again. For safe measure, this should also be done before creating the Connectivity Map and Deployment Profile.

4.5 Using the Connectivity Map Editor

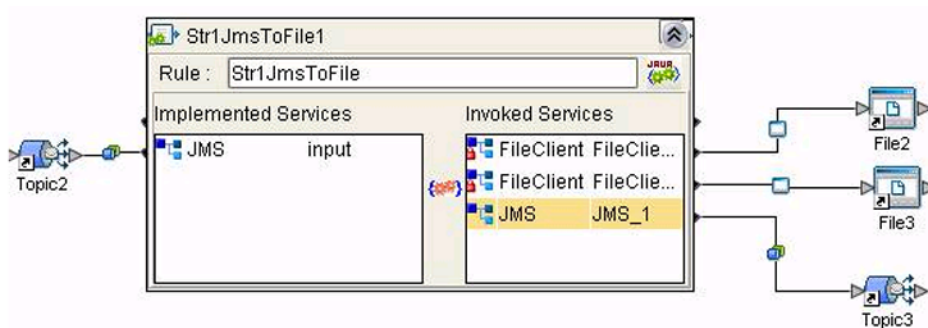
When you create a new Connectivity Map in the Enterprise Explorer, the editor panel displays the Connectivity Map Editor (see Figure 24). To define your Project, you simply drag icons from the toolbar to the workspace, or canvas, to populate the Connectivity Map with the necessary components. You subsequently link the components by dragging the cursor from one to the other.

Figure 24 Connectivity Map Window



When there are multiple destinations, as with a JMS topic, the Connectivity Map Editor cannot resolve which output port to connect to which destination. Because of this, the Collaboration definition must be created first, and the connections must be drawn by opening the Collaboration Binding box in Connectivity Map (see Figure 25).









Figure 25 Linking JMS Topics



4.5.1 Connectivity Map Editor Toolbar

Each toolbar icon in the Connectivity Map Editor is described in Table 17.

Table 17 Connectivity Map Toolbar Icons

Icon	Component	Function
	Service	A logical component that provides the framework for a process or Collaboration. See Service Component on page 54.
	Queue	A Message Destination that conforms to the point-to-point messaging paradigm, having one sender and one receiver. See the <i>eGate Integrator JMS Reference Guide</i> for information.
	Topic	A Message Destination that conforms to the publish/subscribe messaging paradigm, having one sender (publisher) and multiple receivers (subscribers). See the <i>eGate Integrator JMS Reference Guide</i> for information.
	Web Service External Application	Represents a third-party Web service application external to eGate. See Example Web Services Project on page 65.
	External Applications	Represents an application external to eGate. Click the arrow beside the icon to view a list of specific applications to which you can connect. See External Application Drop-Down Menu on page 54.
	File	Represents an external file. See the <i>File eWay Intelligent Adapter User's Guide</i> for more information about the eWay used to read or write the file.
	Scheduler	Represents a scheduling component of the Connectivity Map. Use this component to set data transfer to occur at set intervals. See Scheduler Component on page 55.
	Work List Viewer	See the <i>eInsight Enterprise Service Bus User's Guide</i> for information.

It is important to understand that the logical components appearing in the Connectivity Map are essentially *placeholders* that refer to the “actual” components that exist in the Repository and appear in the Project Explorer. Renaming or deleting a queue or topic in the Connectivity Map only affects the placeholder, not the object in the Repository.

Also, renaming or deleting a queue or topic in the Repository will not affect the existence or name of the associated placeholder in the Connectivity Map. The change will, however, be reflected in the *tooltips* for the placeholder. This allows you to re-assign the placeholder without disrupting the continuity of the Connectivity Map.

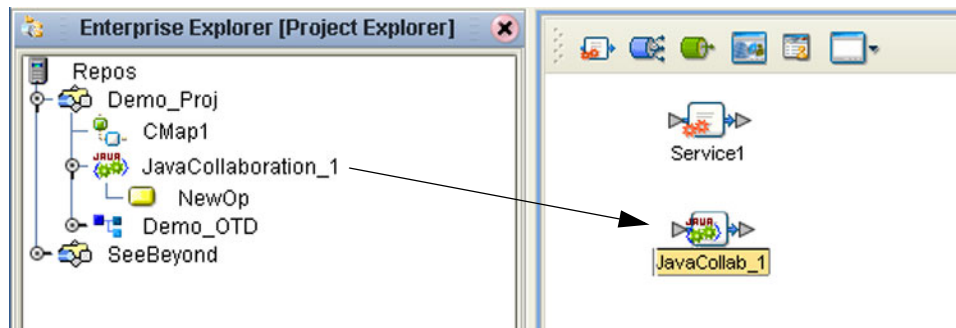
4.6 Creating and Configuring Logical Components

The drag-and-drop components include Services, queues, topics, schedulers, and external applications. Additional components, such as eWays and JMS Clients, are placed automatically when you link the components you have placed manually.

4.6.1 Services

A Service provides a framework for a process or a Collaboration, which contains the information required to execute a set of business rules. Dragging a Collaboration from the Project Explorer to the Service icon in the Connectivity Map defines the Service as a Collaboration.

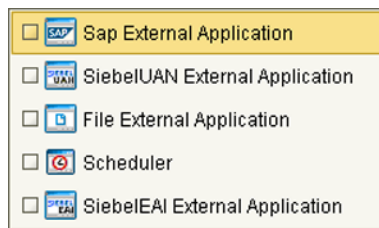
Figure 26 Service Component



4.6.2 External Applications

External Applications are logical representations of external software applications that are being integrated by the eGate system. These are linked to a Service by means of an eWay. Clicking the drop-down arrow beside the External Application icon displays a menu showing those applications corresponding to eWays that have been purchased and installed, plus the Scheduler (see Figure 27).

Figure 27 External Application Drop-Down Menu



Checking the individual External Applications adds the selected icon to the toolbar (see Figure 28).

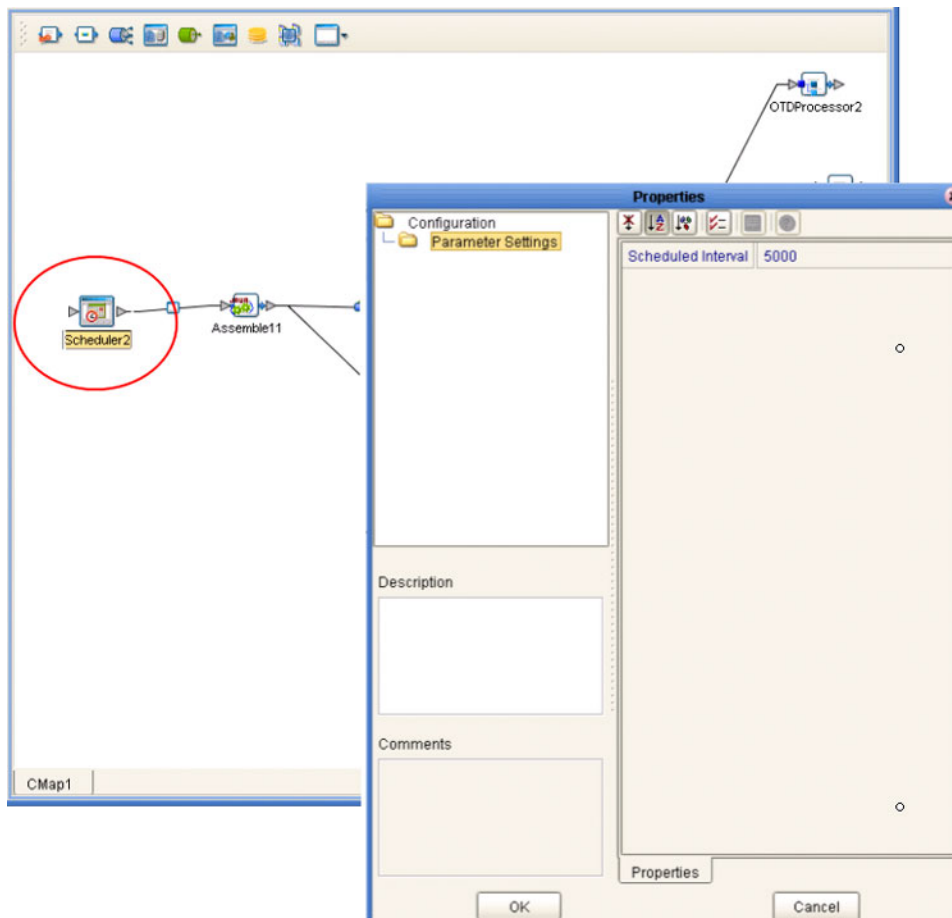
Figure 28 Selected External Applications in Toolbar



4.6.3 Schedulers

A Scheduler allows a service to be performed at a prescribed interval. The interval can be static, or can be made dynamic by using a Project variable for the interval value.

Figure 29 Scheduler Component

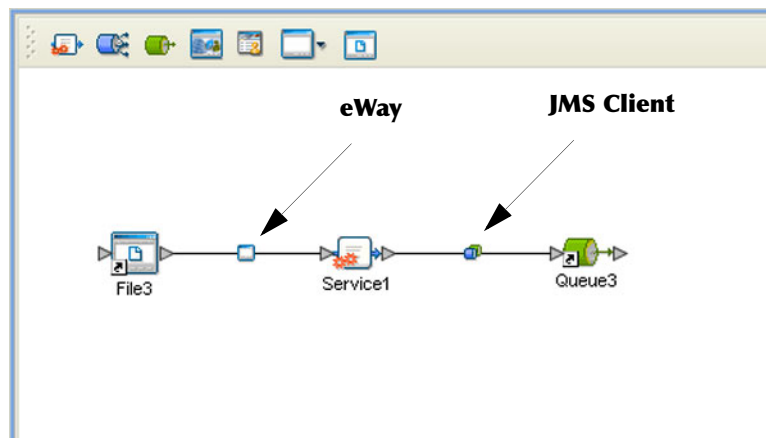


As an example, depicted in Figure 29, a scheduler can be used to invoke a Collaboration with an interval of 5 seconds (5000 milliseconds).

4.7 Component Connections

When you link two components on a Connectivity Map, the Enterprise Designer places either an eWay or JMS Client connection icon on the link, depending upon the type of components you are linking (see Figure 30).

Figure 30 Connection Icons in a Connectivity Map



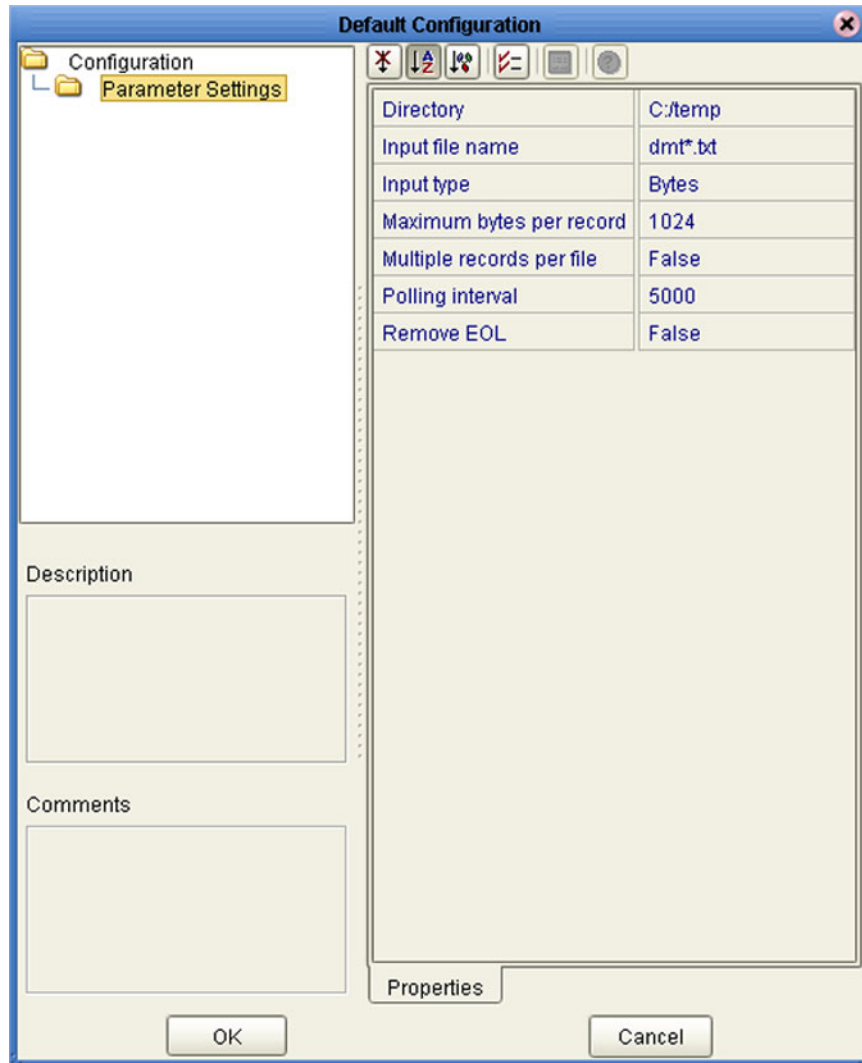
- When you link an external application with a Collaboration, the Enterprise Designer automatically adds an eWay Connection icon to the link. The eWay enables communication and movement of data between the external application and the eGate system. The eWay configuration specifies the logical connection properties for the link. See the individual eWay Intelligent Adapter User's Guides for specific information.
- When you link a Service with a Message Destination (queue or topic), the Enterprise Designer adds a JMS Client Connection icon. The JMS Client configuration specifies the logical connection properties for the linked Service. See the *eGate Integrator JMS Reference Guide* for information.

Note: You must configure the JMS properties, since no default values are assigned. Failure to do so will result in an error at a later time.

4.7.1 Configuring a Connection

Double-clicking an eWay or JMS Client properties handle icon in the Connectivity Map displays the Default Configuration dialog box. As an example, Figure 31 shows a dialog box that lists the configuration properties for a File eWay.

Figure 31 Default Configuration Dialog Box









Note: The first time you double-click an eWay or JMS Client icon, you will see a Templates dialog box. Here, you must designate an eWay to be inbound or outbound, or a Service to be a publisher or subscriber. Clicking OK will then display the Default Configuration dialog box.

The constituent parts of the Default Configuration dialog box are:

- The **Configuration Tree** includes folders that contain configuration and connection properties for the selected eWay or message destination.
- The **Editor Toolbar** contains a series of buttons used to sort and modify the information listed in the Properties folder, as described in Table 18.
- The **Properties** folder lists the default properties for the selected eWay or JMS Client. See, respectively, the individual eWay Intelligent Adapter User’s Guides and the *eGate Integrator JMS Reference Guide* for information.
- The **Description** box contains a brief description of the contents of the item currently selected in the Configuration Tree.
- The **Comments** box lists additional information about the item selected in the Configuration Tree.

Table 18 Default Configuration Toolbar Buttons

Button	Command	Function
	Unsorted	Displays configuration properties in their default order.
	Sort by Name	Sorts configuration properties alphabetically by name.
	Sort by Type	Displays configuration properties by property type.
	Show Editable Properties Only	Displays only the properties of an eWay or message destination that can be modified.
	Customizer	Displays the Customizer dialog box, which you can use to customize the selected eWay or message destination.
	Help	Displays the online help documentation for the Configuration Editor.

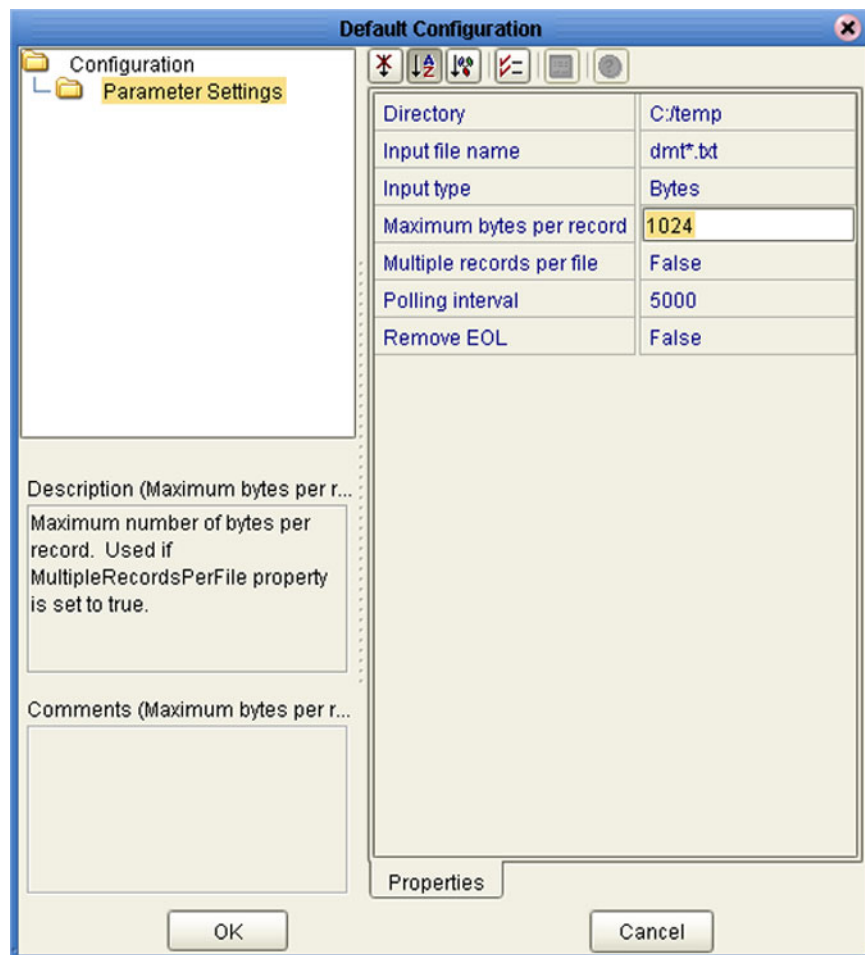
4.7.2 Modifying a Configuration Property

The properties initially listed in the Configuration Properties folder are the default settings for the selected eWay or JMS Client. This section describes how to change the default configuration values, using an inbound eWay as an example.

To change a configuration property

- 1 Double-click on an eWay or JMS Client properties handle icon in a Connectivity Map to display the Default Configuration dialog box.
- 2 Click on the right column of a row to place the row in edit mode. For example, click on the *Maximum bytes per record* field as shown in Figure 32, which highlights the current value.

Figure 32 Editing a Default Configuration Field



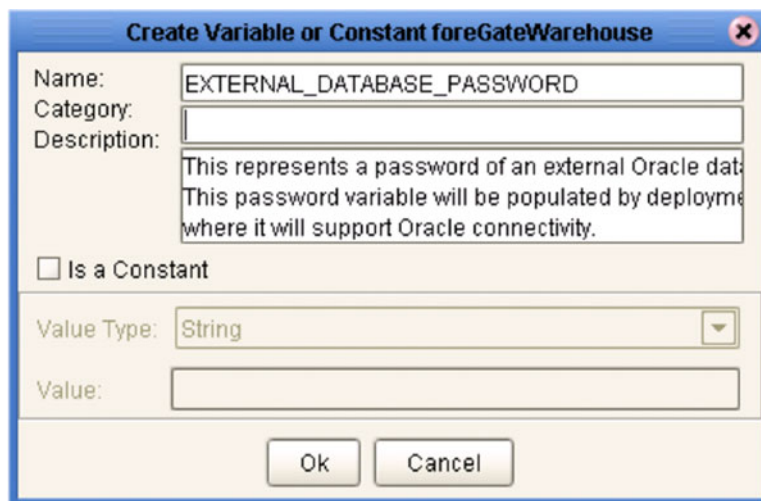
- 3 Enter a new value in the field. (Note that in this example, the parameter is not used unless you allow multiple records per file, as explained in the Description box).
- 4 Click **OK** to save the new settings.

4.8 Defining Constants and Variables

You can define variables and constants for a specific Project. Variables function as placeholders, having values that are determined when you create a specific Deployment Profile (see [Mapping Variables](#) on page 98). Project variable values can be literals or Environmental constants.

For example, a project variable is defined to represent a password of a database user in a target environment (see Figure 33). System managers will assign an actual value to this variable in the deployment profile editor. The value of the assigned project variable—an Environment constant—is then used to connect the database in the target environment.

Figure 33 Project Variable Creation



The screenshot shows a dialog box titled "Create Variable or Constant foreGateWarehouse". The "Name" field contains "EXTERNAL_DATABASE_PASSWORD". The "Category" field is empty. The "Description" field contains two lines of text: "This represents a password of an external Oracle dat..." and "This password variable will be populated by deployme..." followed by "where it will support Oracle connectivity." on a new line. There is an unchecked checkbox labeled "Is a Constant". Below this is a "Value Type" dropdown menu set to "String" and an empty "Value" text box. At the bottom are "Ok" and "Cancel" buttons.

A constant is a name-value pair; when you create a constant you assign a permanent value to it, which cannot be overridden. An example of this would be a standard currency used globally throughout the system (see Figure 34).

Figure 34 Project Constant Creation

The dialog box is titled "Create Variable or Constant foreGateWarehouse". It contains the following fields and options:

- Name: CONSTANT_STANDARD_CURRENCY
- Category: (empty)
- Description: This constant represents a standard currency that is used throughout the entire system.
- Is a Constant
- Value Type: String
- Value: USD
- Buttons: Ok, Cancel

These constants and variables are automatically added to a Variables and Constants object group within the Project (see Figure 35).

Figure 35 Variables and Constants Object Group

The screenshot shows the SeeBeyond Enterprise Designer 5.0 interface. The main window is titled "SeeBeyond Enterprise Designer 5.0 - Variables and Constants [eGateWarehouse_Variables]". The "Enterprise Explorer" on the left shows a tree view of the project structure, with "eGateWarehouse" selected. The main area displays the "eGateWarehouse: Variables&Constants" object group as a table:

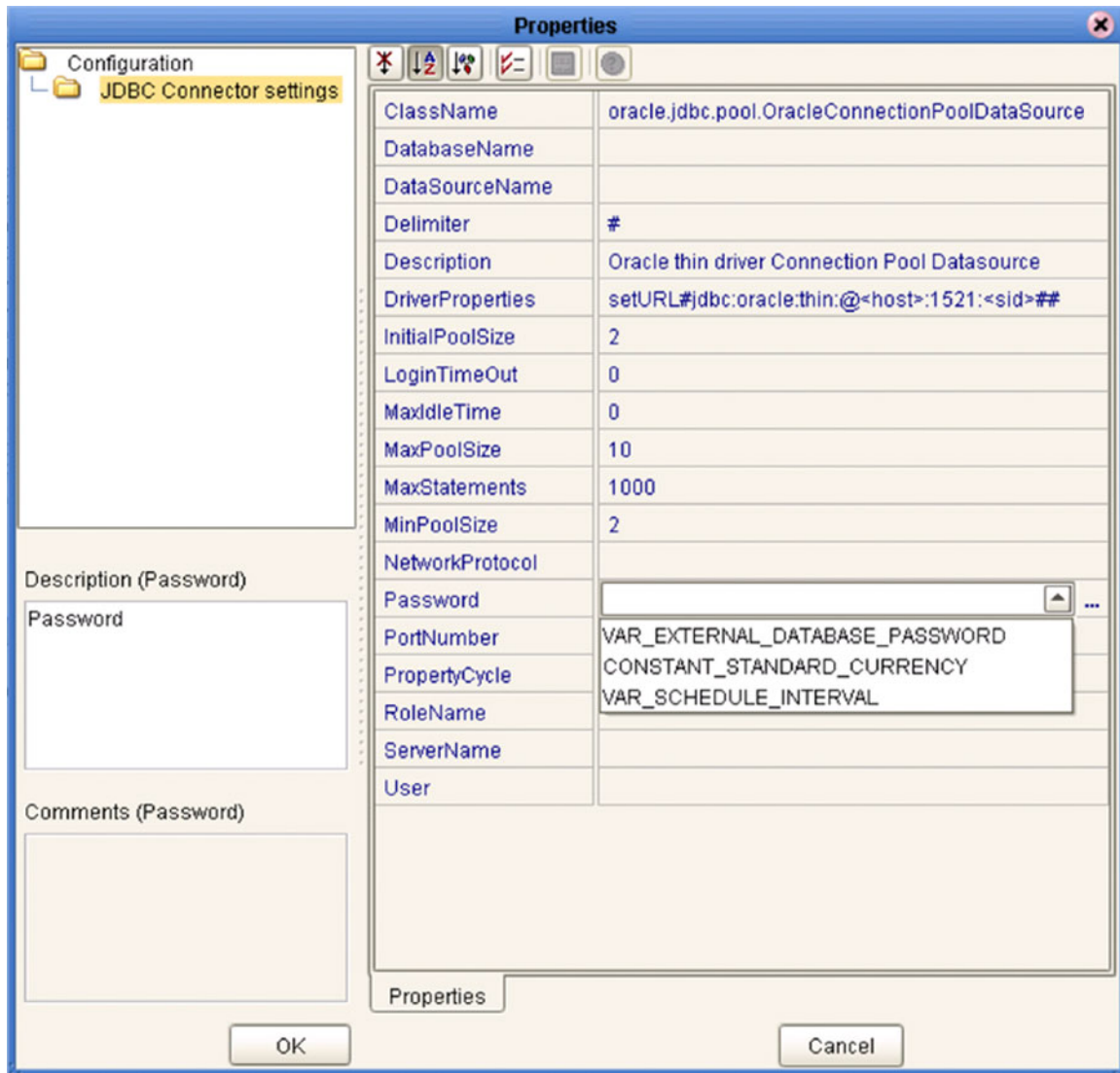
Name	Value	Constant	Category	Description
CONSTANT_STANDARD_CURRENCY	USD	<input checked="" type="checkbox"/>		This constant represent a standard currency that is used throughout the ent...
VAR_EXTERNAL_DATABASE_PASSWORD		<input type="checkbox"/>		This represents a password of an external Oracle database user. This pas...

Below the table is a section titled "Add a New Variable or Constant" with the following fields and options:

- Name: (empty)
- Category: (empty)
- Description: (empty)
- Is a Constant
- Value Type: String
- Value: (empty)
- Button: Ok

The Project variables and constants can be referenced as properties within the Project. For example, the password variable described previously can be selected from the Variables and Constants object group to provide the Password property in the JDBC Connector settings dialog box (see Figure 36).

Figure 36 Connector Properties



See [Mapping Variables](#) on page 98 for information on setting the values for Project variables.

4.9 Web Services

In general, all ICAN objects that expose themselves as a Web Service (such as an eInsight business process) are presented in the SeeBeyond UDDI Repository (see Figure 37). The URL of this repository is:

```
http://ICAN_Suite_host_name:enterprise_manager_installation_port/  
stcuddi
```

Figure 37 SeeBeyond UDDI Repository



Environment	Service Name	WSDL
Environment2	BusinessProcess1	http://art2k:10000/repository/MyRepository/data/uddocs/Environment2/BusinessProcess1/BusinessProcess1.wsdl
Environment2	BusinessProcess1	http://art2k:10000/repository/MyRepository/data/uddocs/Environment2/BusinessProcess1/BusinessProcess2.wsdl

Each entry in the UDDI Repository includes:

- The ICAN environment name.
- The actual (Web) Services name.
- The location of the Web Service’s WSDL file.

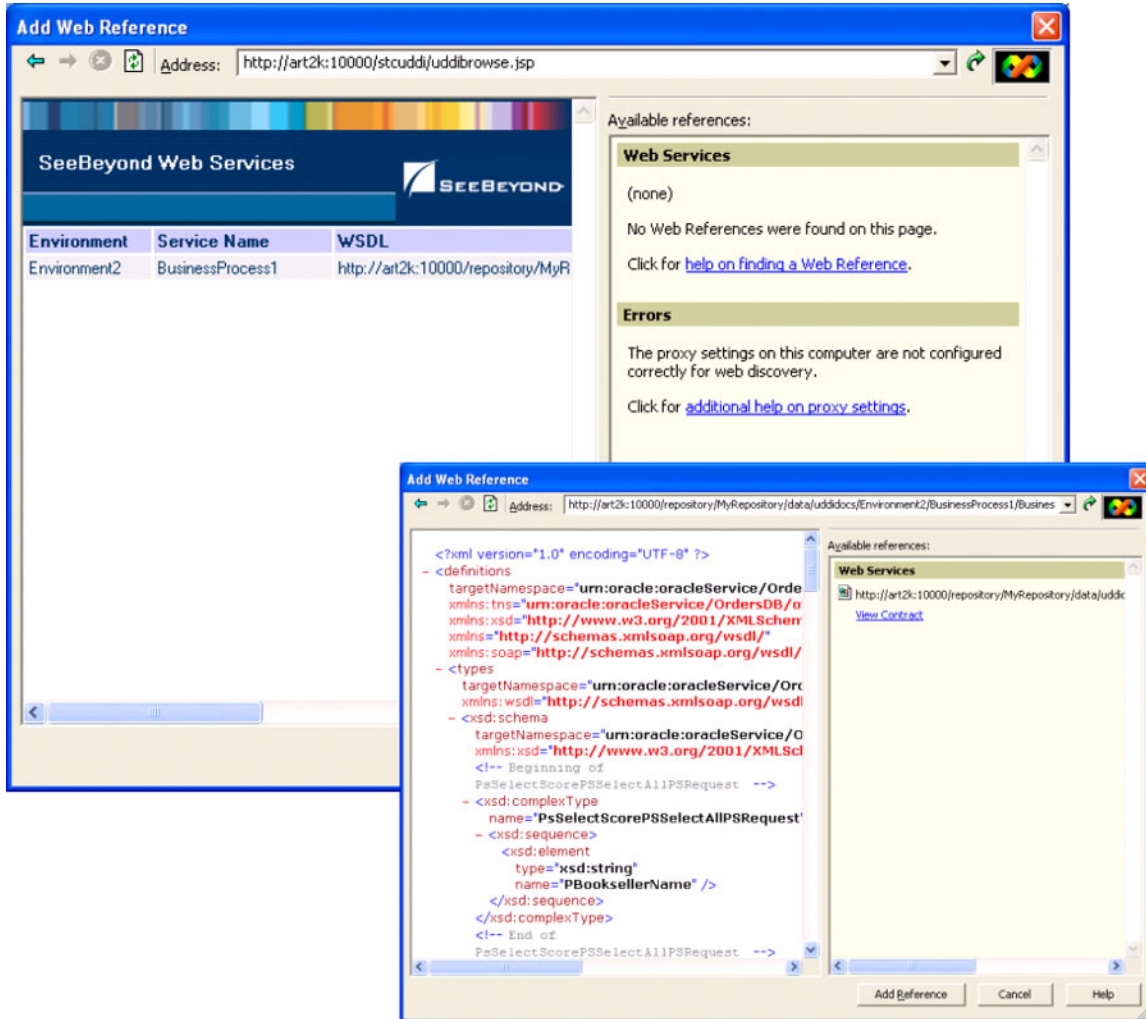
By selecting an entry its WSDL file is displayed, as shown in Figure 38.

Figure 38 Example Web Service WSDL File

```
<?xml version="1.0" encoding="UTF-8" ?>
- <definitions targetNamespace="urn:oracle:oracleService/OrdersDB/otdGetCreditScore" xmlns:tns="urn:oracle:oracleService/OrdersDB/otdGetCreditScore"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/" xmlns:soap11="http://schemas.xmlsoap.org/wsdl/soap11/" />
- <types targetNamespace="urn:oracle:oracleService/OrdersDB/otdGetCreditScore" xmlns:tns="urn:oracle:oracleService/OrdersDB/otdGetCreditScore" />
- <xsd:schema targetNamespace="urn:oracle:oracleService/OrdersDB/otdGetCreditScore" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <!-- Beginning of PsSelectScorePSSelectAllPSRequest -->
  - <xsd:complexType name="PsSelectScorePSSelectAllPSRequest">
    - <xsd:sequence>
      <xsd:element type="xsd:string" name="PBooksellerName" />
    </xsd:sequence>
  </xsd:complexType>
  <!-- End of PsSelectScorePSSelectAllPSRequest -->
  <!-- Beginning of PsSelectScorePSSelectOnePSResponseType -->
  - <xsd:complexType name="PsSelectScorePSSelectOnePSResponseType">
    - <xsd:sequence>
      <xsd:element type="xsd:decimal" name="CREDIT_SCORE" />
    </xsd:sequence>
  </xsd:complexType>
  <!-- End of PsSelectScorePSSelectOnePSResponseType -->
  <!-- Beginning of PsSelectScorePSSelectMultiplePSResponseType -->
  - <xsd:complexType name="PsSelectScorePSSelectMultiplePSResponseType">
    - <xsd:sequence>
      <xsd:element type="xsd:long" name="rowCount" />
      <xsd:element type="xsd:decimal" name="CREDIT_SCORE" />
    </xsd:sequence>
  </xsd:complexType>
  <!-- End of PsSelectScorePSSelectMultiplePSResponseType -->
  <!-- Beginning of PsSelectScorePSSelectAllPSResponseType -->
  - <xsd:complexType name="PsSelectScorePSSelectAllPSResponseType">
    - <xsd:sequence>
      <xsd:element type="xsd:long" name="rowCount" />
    </xsd:sequence>
  </xsd:complexType>
  </xsd:schema>
```

The SeeBeyond UDDI Repository can be used in a third party tool, for example Microsoft Visual Studio (see Figure 39). In this example, a so-called *Web reference* (to the UDDI repository) is added to a C# project.

Figure 39 Microsoft Visual Studio Example



eGate Integrator can exchange data with Internet and Web Services applications using the Web Services Description Language (WSDL). This language is XML-based and is used to define Web services and describe how to access them. The WSDL OTD Wizard is used to build OTDs that are used in the Project Collaborations (see [WSDL-Based OTDs](#) on page 73).

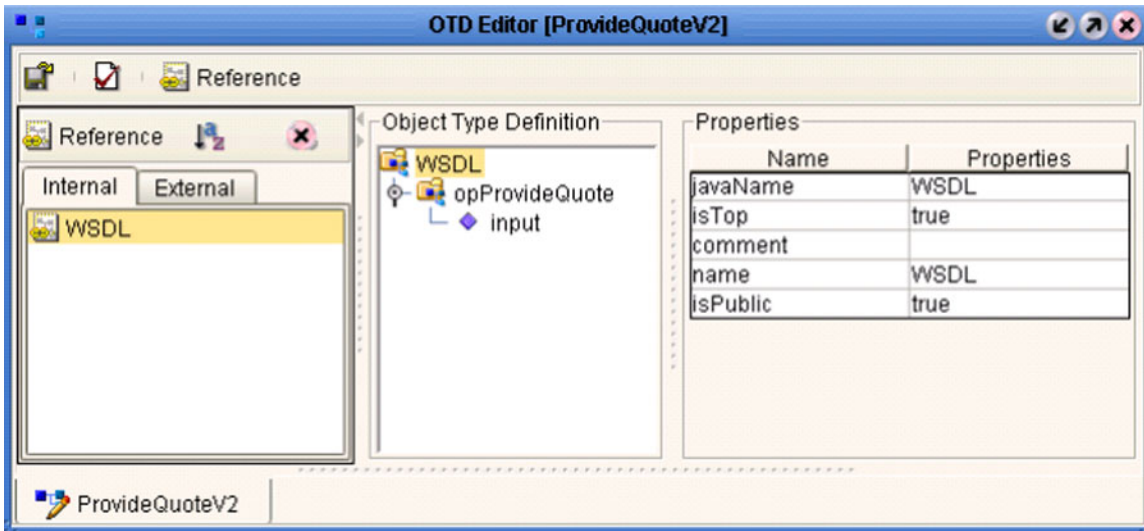
4.9.1 Example Web Services Project

As an illustrative example, we will show a project called "prgProvideStockQuote".

Creating the OTDs

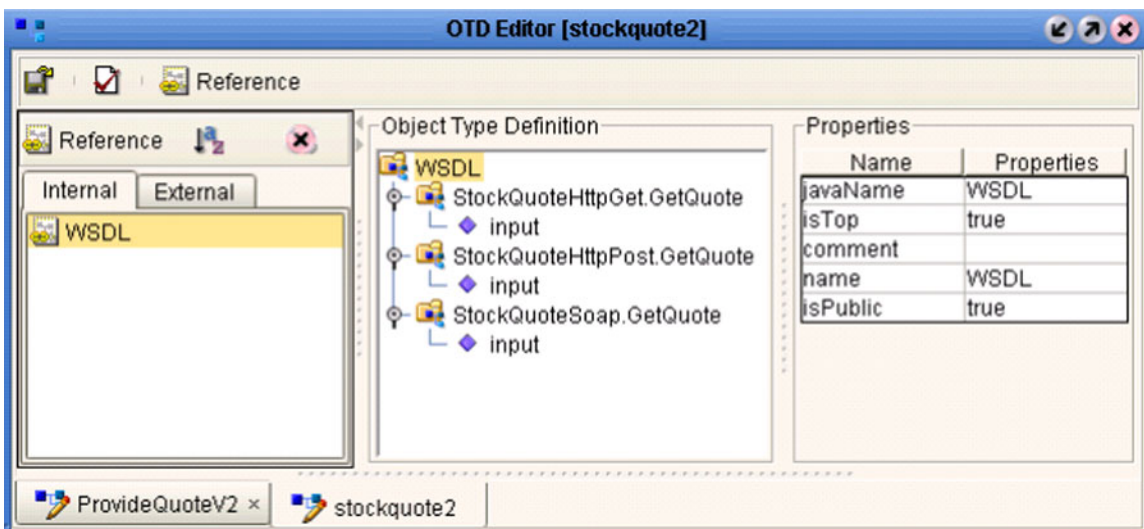
Three OTDs are required for this Project. First is an OTD based on the WSDL called **ProvideQuoteV2.wsdl** (see Figure 40).

Figure 40 Web Service Example (OTD 1)



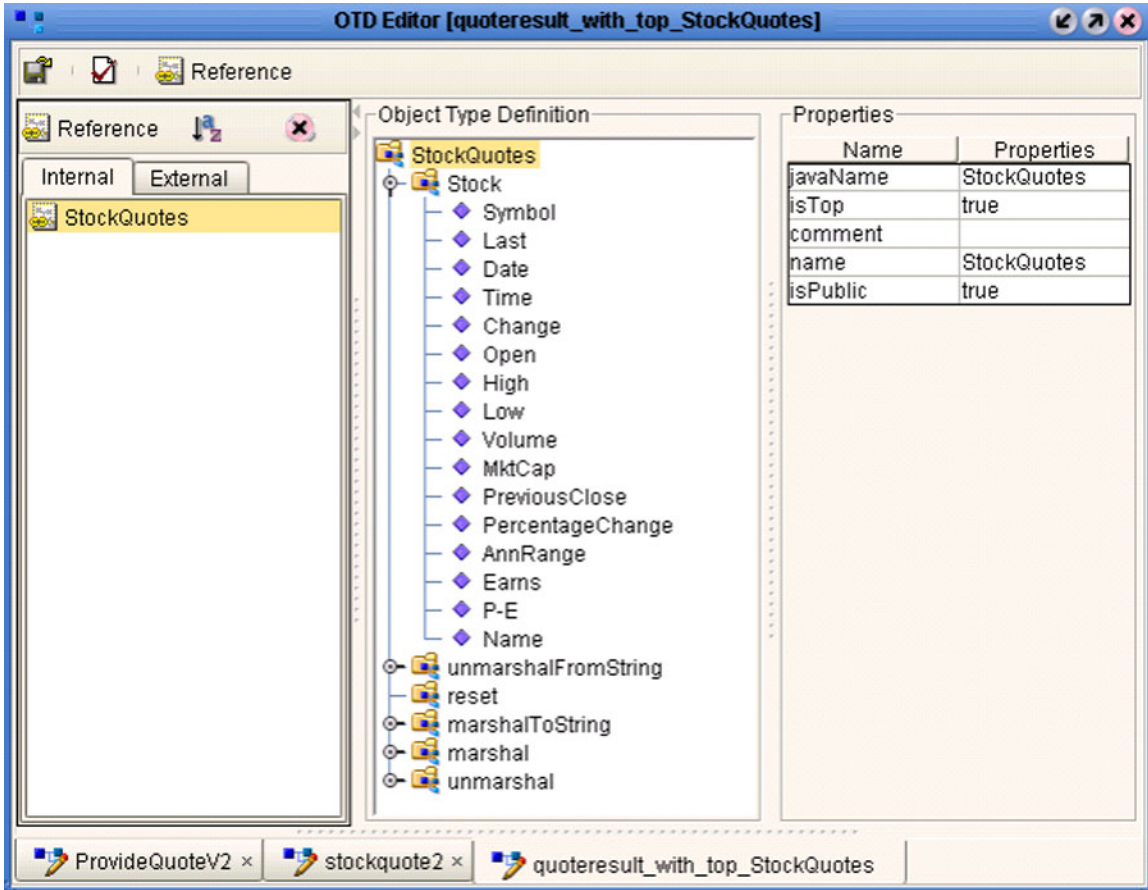
Second is an OTD based on the WSDL for calling the external stock quote Web Service, based on the **stockquote2.wsdl** (see Figure 41).

Figure 41 Web Service Example (OTD 2)



Third is an OTD for unmarshaling the result we'll be receiving from the external Web Service. The result is in XML and a DTD file has been provided for the Web Service called **quoteresult.dtd** (see Figure 42).

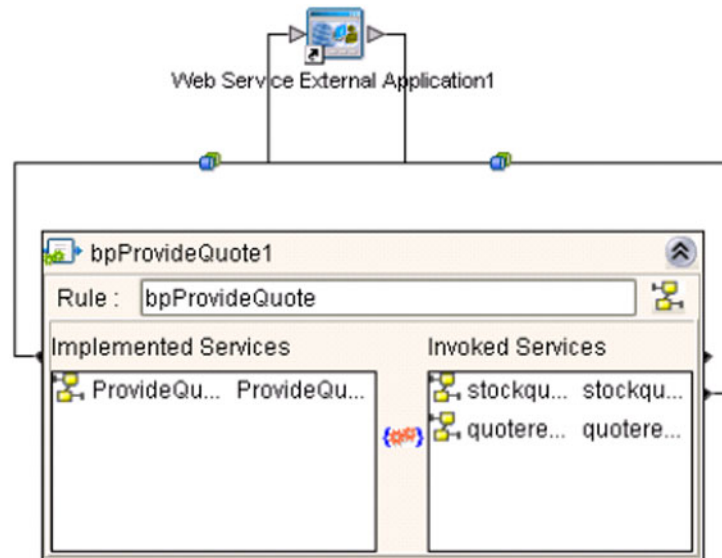
Figure 42 Web Service Example (OTD 3)



Using the Connectivity Map

In this example, an eInsight business process is placed into the eGate Connectivity Map, and a Web Service External Application is added to the business process (see Figure 43).

Figure 43 Web Service Example (Connectivity Map)



The resulting WSDL file is shown in Figure 44.

Figure 44 Web Service Example (WSDL File)

```
<?xml version="1.0" encoding="UTF-8" ?>
- <definitions targetNamespace="http://seebeyond/quoteservice" xmlns:tns="http://seebeyond/quoteservice"
  xmlns:sbypnx="http://bpel.seebeyond.com/hawaii/5.0/privateExtension/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
- <types targetNamespace="http://seebeyond/quoteservice">
  - <xsd:schema targetNamespace="http://seebeyond/quoteservice"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element type="xsd:string" name="QuoteRequest" />
    <xsd:element type="xsd:string" name="QuotePrice" />
    <xsd:element type="xsd:string" name="QuoteCompany" />
    </xsd:schema>
  </types>
- <message name="msgProvideQuoteRequest">
  <part name="TICKER" element="tns:QuoteRequest" />
</message>
- <message name="msgQuoteResponse">
  <part name="price" element="tns:QuotePrice" />
  <part name="company" element="tns:QuoteCompany" />
</message>
- <portType name="ptProvideQuote">
  - <operation name="opProvideQuote">
    <input message="tns:msgProvideQuoteRequest" />
    <output message="tns:msgQuoteResponse" />
  </operation>
</portType>
- <binding name="ptProvideQuoteBinding" type="tns:ptProvideQuote">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
  - <operation name="opProvideQuote">
    <soap:operation soapAction="{http://seebeyond/quoteservice}ptProvideQuote::opProvideQuote" />
    - <input>
      <soap:body use="literal" />
    </input>
    - <output>
      <soap:body use="literal" />
    </output>
  </operation>
</binding>
- <service name="bpProvideQuoteService">
  - <port name="ptProvideQuoteBindingService" binding="tns:ptProvideQuoteBinding">
    <soap:address location="http://localhost:18003/WebServiceExt/service/ptProvideQuote" />
  </port>
</service>
</definitions>
```

Upon activation, the deployed WSDL file is displayed in the UDDI Repository as shown in Figure 45.

Figure 45 Web Service Example (UDDI Repository)



Environment	Service Name	WSDL
Environment1	bpProvideQuote	http://localhost:12000/repository/Repository/data/uddidocs/Environment1/bpProvideQuote/bpProvideQuote.wsdl

Object Type Definitions

This chapter describes the OTD creation process. The Enterprise Designer includes two tools, the OTD Wizard and OTD Editor, to help you create and customize OTDs.

5.1 Overview

An Object Type Definition (OTD) is a description of a complex hierarchical data structure that can be accessed and manipulated by your code in a Collaboration. OTDs typically have a specific external representation format that is used to store and transport the OTD contents through the parts of a eGate Project. The OTD defines both the run-time structure and the external representation.

At run time, an OTD instance is accessed from BPEL using XPath expressions. Typically, a collaboration will receive a message containing the external representation of a particular OTD. It will use the *unmarshal* method of an instance of that OTD to parse the data and make it accessible through the hierarchical data structure. Then it will perform some operation: for example, copying parts of the data to another OTD instance. Finally, it will invoke the *marshal* method on the other OTD instance to render the contents of its data structure as a single, serialized data stream for further transport.

5.1.1 OTD Types

Externally-Defined OTDs

Externally-defined OTDs are based on formats or standards external to eGate Integrator, such as Web Services Definition Language (WSDL), XML Schema Definition (XSD), and various proprietary formats such as SAP BAPI. Some of these OTDs are *messagable*, others are API-based. Externally defined OTDs are read-only.

5.1.2 Building OTDs

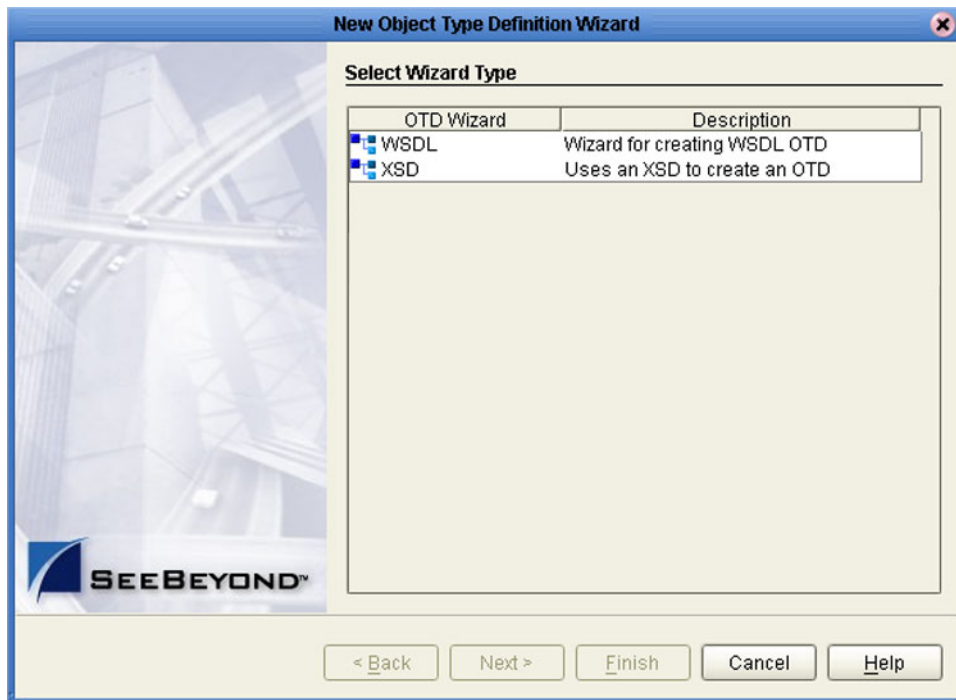
Wizards are provided in the Enterprise Designer GUI to guide you through the OTD building process. These Wizards call back-end builders that actually implement the building of the code, based on the provided information.

5.2 Using the OTD Wizard

Right-click on a Project in the Enterprise Explorer to display the Project context menu, then select **New Object Type Definition** to display the OTD Wizard, shown in Figure 46. The initial dialog allows you to select a specific OTD Wizard. These Wizards are respectively described in:

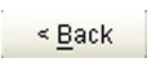



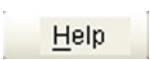
- [Using the WSDL Wizard](#) on page 73
- [Using the XSD Wizard](#) on page 78

Figure 46 OTD Wizard Selection Dialog



5.2.1 Navigation Buttons

Table 19 OTD Wizard Navigation Buttons

Button	Function
	Returns to the previous step in the wizard. This button is disabled on the first step.
	Goes to the next step in the wizard. This button is disabled on the last step.
	Saves all OTD settings and closes the wizard. This button is only enabled on the last step.
	Closes the wizard without saving the OTD.
	Displays the online help documentation for the OTD Wizard dialog box.

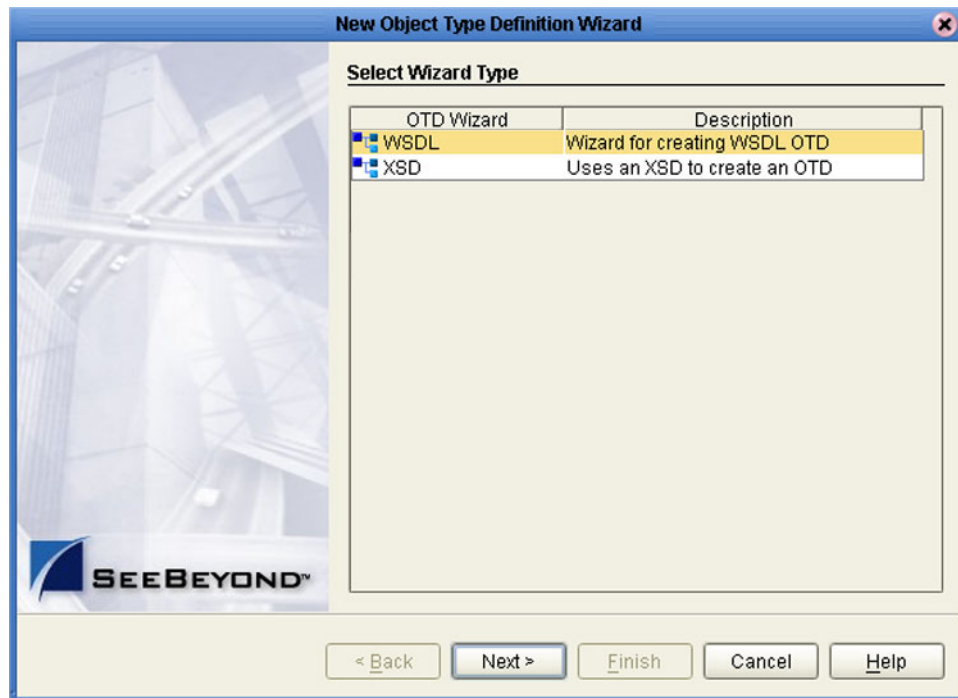
5.3 WSDL-Based OTDs

5.3.1 Using the WSDL Wizard

To create an OTD file from a WSDL file

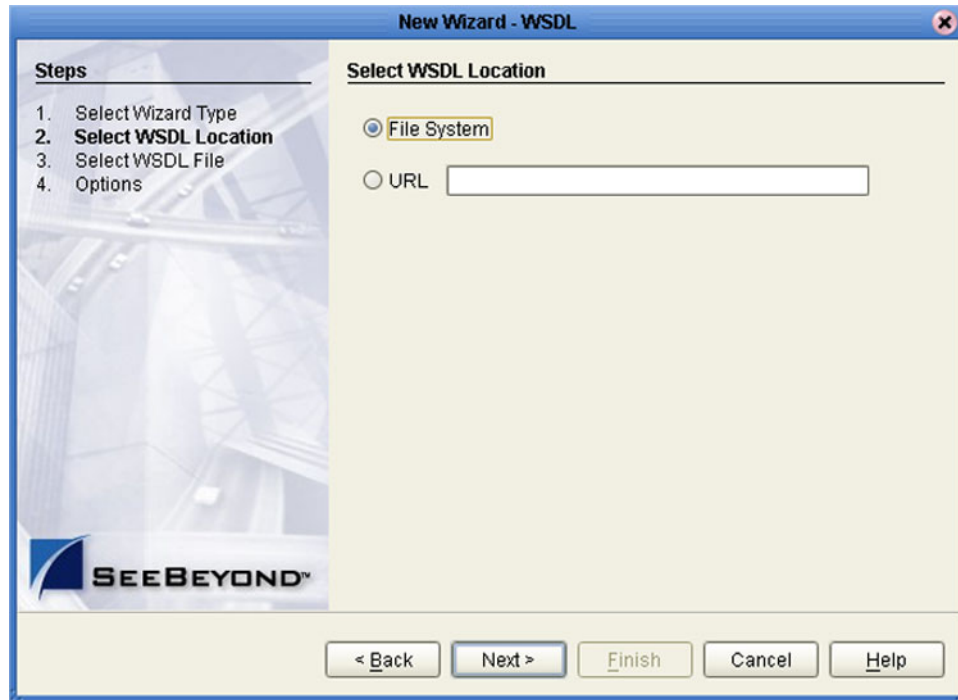
- 1 Select **WSDL** from the *Select an Object Type Definition Wizard* list (see Figure 47) to create an OTD file from an WSFL file.

Figure 47 OTD Wizard Selection: WSDL Wizard



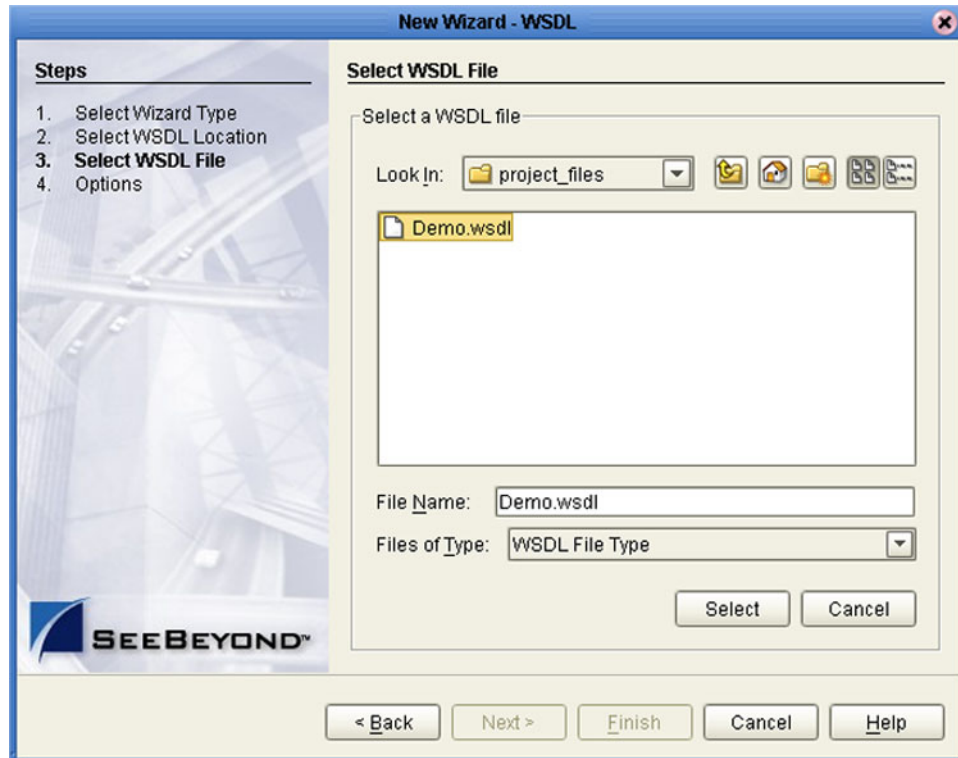
- 2 Click **Next** to display the *Select WSDL File Location* dialog, shown in Figure 48

Figure 48 WSDL Wizard: Select WSDL Location



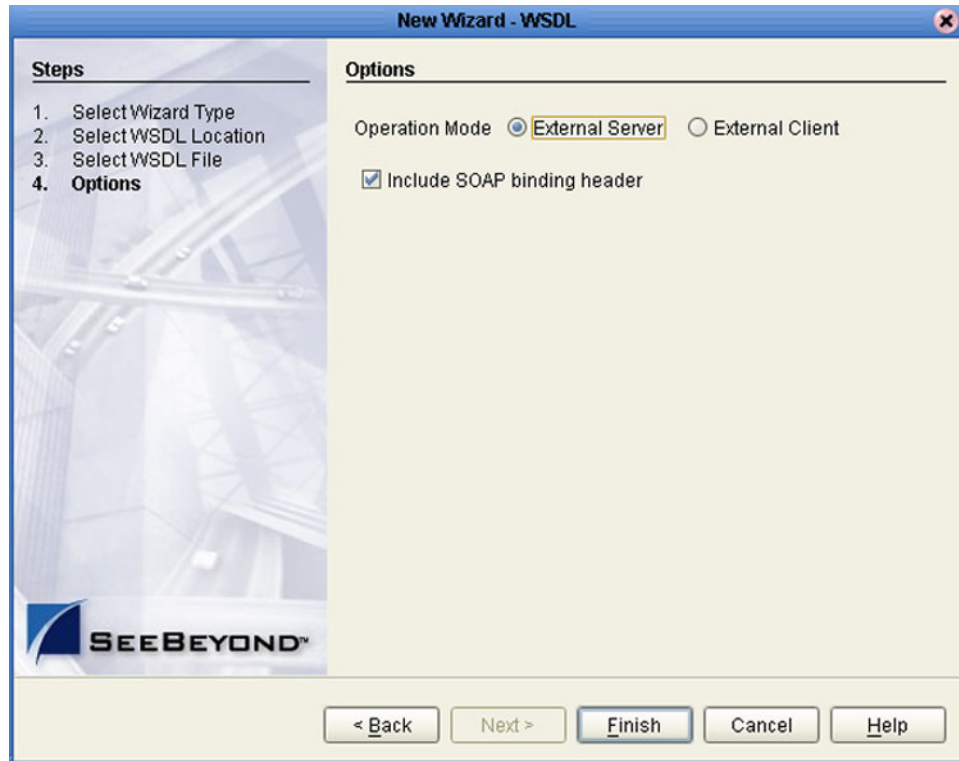
- 3 Select **File System** or enter a **URL**, depending upon where your WSDL file is located.
- 4 Click **Next** to display the *Select WSDL File* dialog, shown in Figure 49.

Figure 49 WSDL Wizard: Select WSDL File



- 5 In the *Look in* drop-down list, navigate to the WSDL file or files that you want to use to create the OTD. Click **Select** to add the files to the *List of Selected WSDLs*.
- 6 Click **Next** to display the *Select OTD Options* dialog, shown in Figure 50.

Figure 50 WSDL Wizard: Select OTD Options



- 7 Select the check boxes next to the OTD options you want to enable, and click **Finish** to add the OTD to the Enterprise Designer with the selected OTD options.

5.3.2 WSDL OTD Operation

WSDL Operation Elements

To tie your messages together as a request-response pair corresponding to a method call, you must define operations using the WSDL **<operation>** element. A WSDL operation specifies which message is the *input* and which message is the *output*.

Inside the WSDL file's **<operation>** element, you specify your **<input>** and **<output>** elements. Each element refers to the corresponding message by its fully qualified name. The collection of all WSDL operations (that is, methods) exposed by your service is called a **portType** and is defined using the WSDL **<portType>** element.

The **<operation>** element is a child of **<portType>**. You can name the **<portType>** whatever you want. The port type **name** attribute provides a unique name among all the PortTypes defined within the enclosing WSDL file. Each WSDL operation is named via the **name** attribute.

Each operation within a WSDL OTD (like its WSDL file counterpart) uses one of the following operation modes for communication:

- **One-way:** The server receives a message from the client; also referred to as “fire and forget.”
- **Request-response:** The server receives a message from the client and sends a correlated message back

WSDL OTD Structure

The WSDL OTD has the following basic structure:

```
Root Node
  PortType_XXX
    Operation_XXX
      Input_XXX
      Output_XXX
  PortType_XXX
    Operation_XXX
      Input_XXX
      Output_XXX (and so on)
```

Where **XXX** is the name for each element given in the original WSDL file.

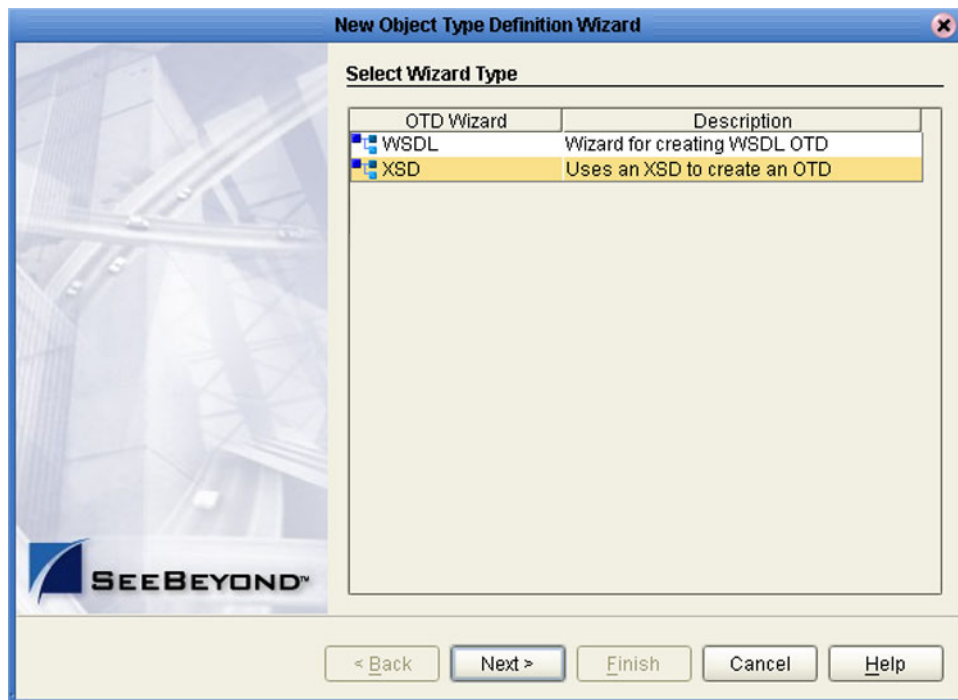
5.4 XSD-Based OTDs

5.4.1 Using the XSD Wizard

To create an OTD file from an XSD file

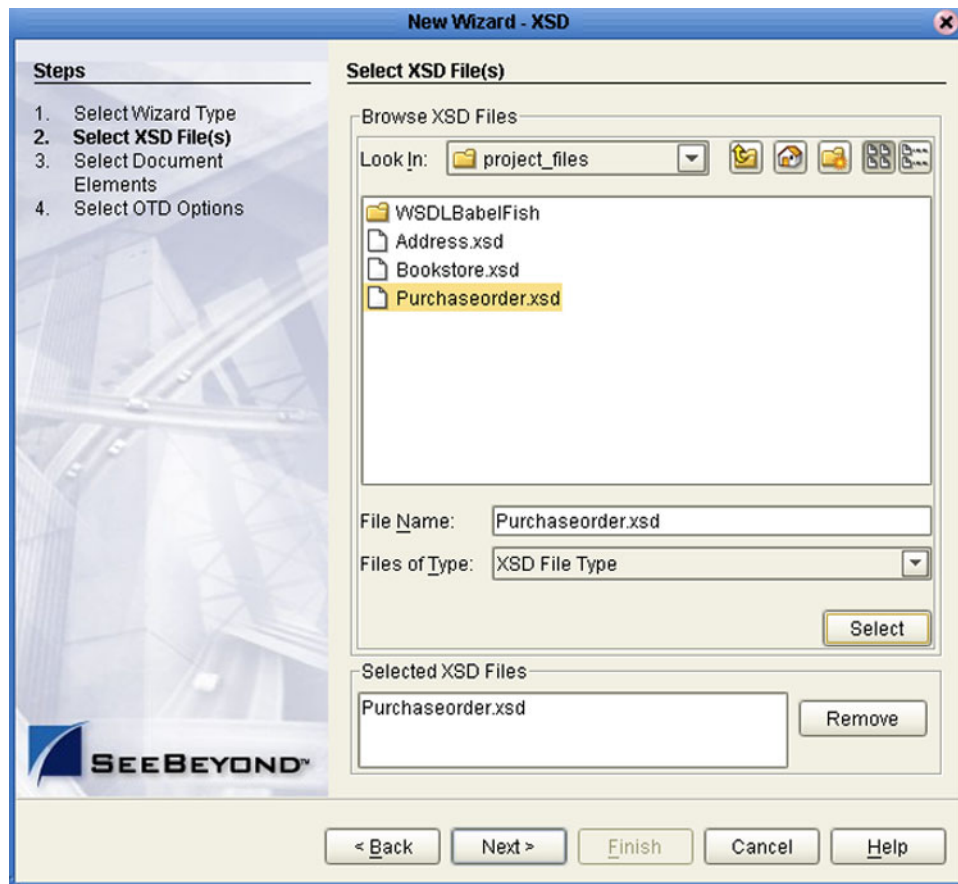
- 1 Select **XSD** from the *Select an Object Type Definition Wizard* list (see Figure 51) to create an OTD file from an XSD file.

Figure 51 OTD Wizard Selection: XSD Wizard



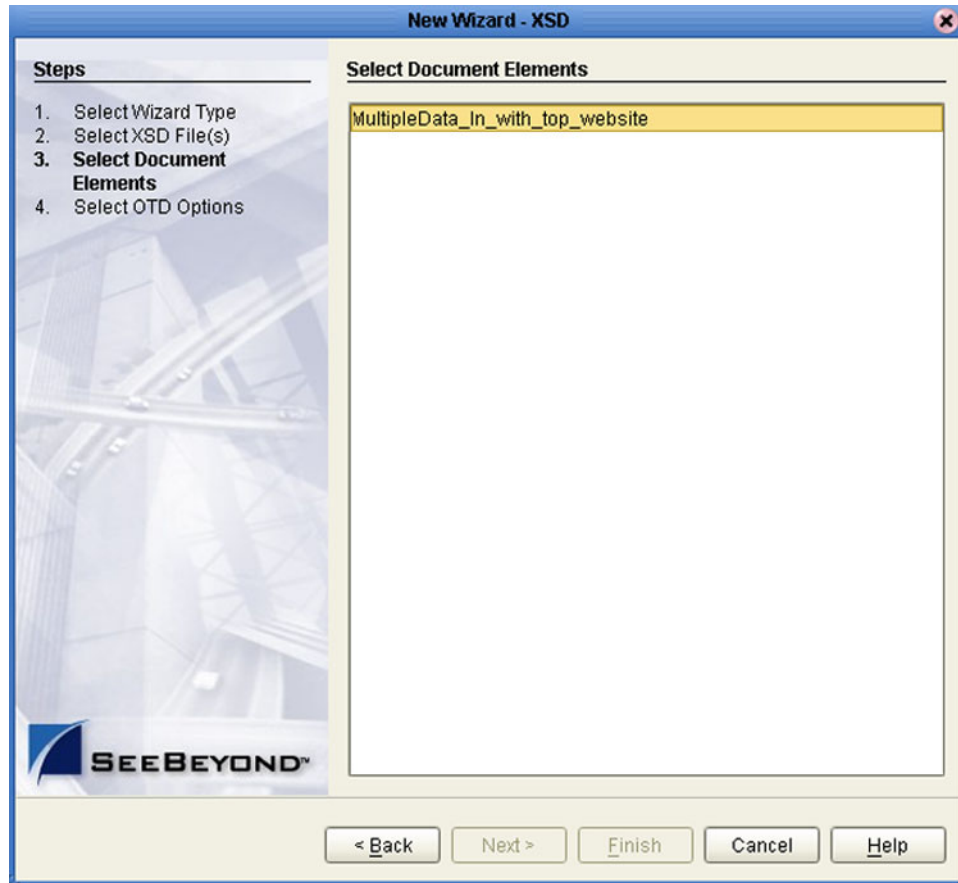
- 2 Click **Next** to display the Select XSD File(s) dialog box, shown in Figure 52.

Figure 52 XSD Wizard: Select XSD File(s)



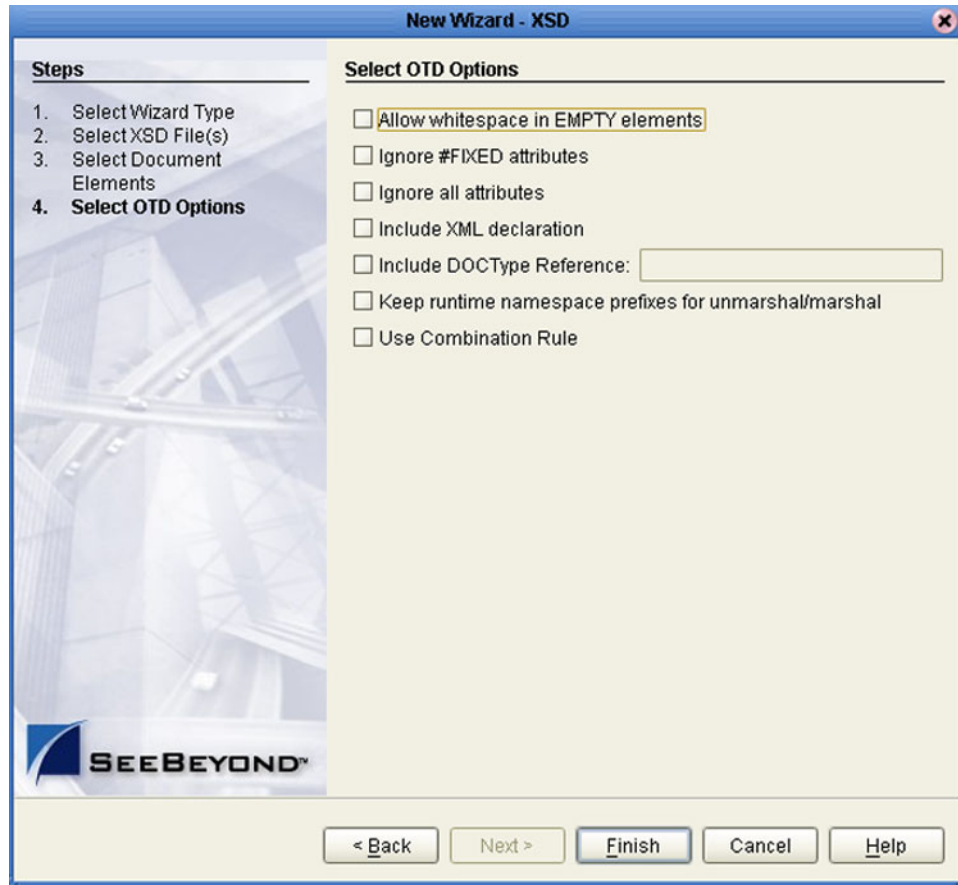
- 3 In the *Look in* drop-down list, navigate to the XSD file or files that you want to use to create the OTD. Click **Select** to add the files to the *List of Selected XSDs*.
- 4 Click **Next** to display the *Select Document Elements* dialog box, shown in Figure 53.

Figure 53 Select Document Elements Dialog Box



- 5 Select the elements of the document that you want to include in the OTD.
- 6 Click **Next** to display the *Select OTD Options* dialog box, shown in Figure 54.

Figure 54 Select OTD Options Dialog Box



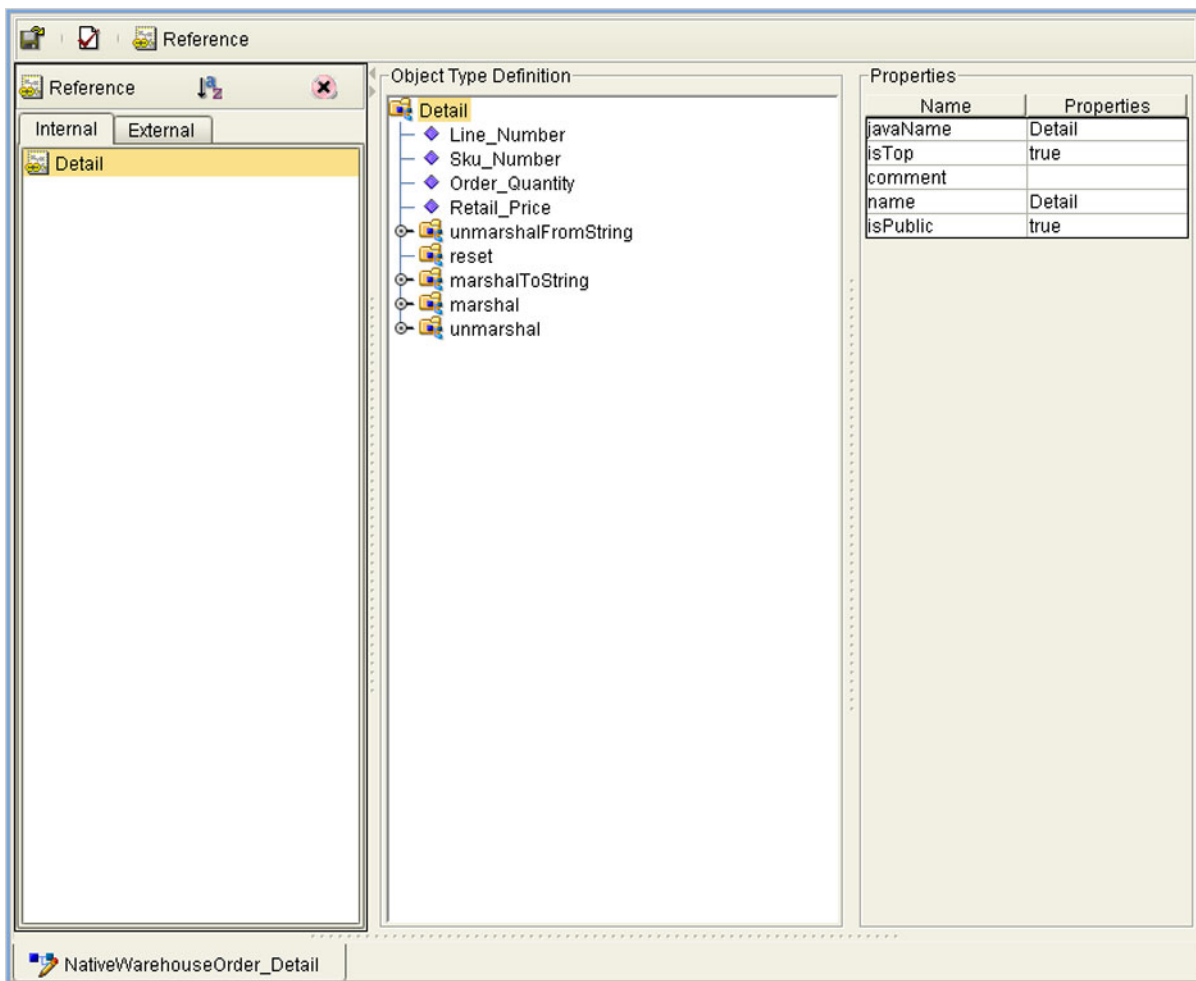
- 7 Select the check boxes next to the OTD options you want to enable, and click **Finish** to add the OTD to the Enterprise Designer with the selected OTD options.

5.5 Using the OTD Editor

After you create an OTD file using the OTD Wizard, the OTD Editor appears in the editor panel of the Enterprise Designer, as shown in Figure 55. You can also invoke the OTD Editor by selecting **Open** in the context menu for an existing OTD in the Project Explorer. OTDs are saved to the Project automatically.

Important: *If you delete an OTD in the Project Explorer, any Collaboration Definitions that have been built using that OTD will be affected. It is recommended that you run the Impact Analyzer before attempting to delete any OTDs (see [Impact Analysis](#) on page 118).*









Figure 55 OTD Editor



Major features of the OTD Editor interface are:

- **Reference**
This area contains internal and external templates for the OTD file.
- **Object Type Definition**
This area displays each field and element included in the OTD file.
- **Properties**
This area displays details about the OTD file or field selected in the *Object Type Definition* list.
- **Tester**
This area displays in the bottom part of the window when you click **Tester**. Use this area to perform tests on the contents of the OTD.
- **Toolbars**
Several toolbars appear in the OTD Editor, containing icons as described in Table 20.

Table 20 OTD Editor Toolbar Icons

Icon	Command	Function
	Save as New Name in Repository	Saves current OTD under a new name in the Repository.
	Tester	Displays/refreshes the Tester area.
	Toggle Reference Tab Panel	Displays/hides the Reference area.
	Sort by Name	Sorts list alphabetically by name.
	Run Tester	Runs the tester with the entered values.
	Open	Displays file browser.
	Save	Saves displayed file.
	Refresh	Repopulates the OTD object elements with the values from the data display panel.

5.5.1 Node Management

The OTD Editor allows you to:

- **Add** nodes and elements to an OTD.
- **Delete** nodes and elements from an OTD.

When a node is *deleted*, both the node and its associated 'children' (data elements) are deleted.

- **Prune** nodes in an OTD.

When a node is *pruned*, only its associated 'children' (data elements) are deleted, while the node itself is preserved. Pruning can only be performed on nodes.

These commands are accessed from the node context menu.

5.6 Using the OTD Tester

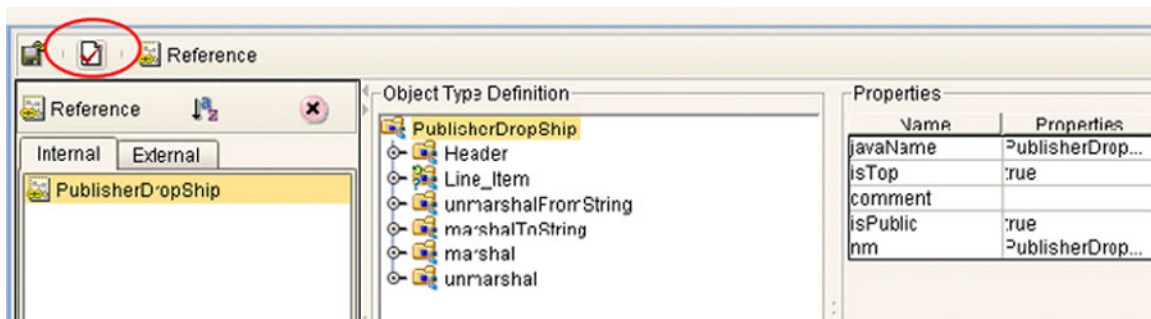
The OTD tester provides a facility to verify the correctness of OTDs, for example to:

- Prevent data errors at runtime.
- Verify that all required data elements are available.
- Verify that all used data formats are correct.

To use the OTD tester

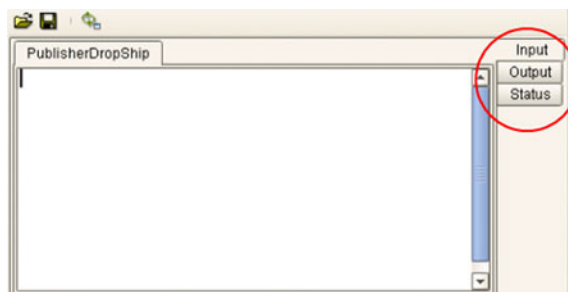
- 1 Open or create an OTD.
- 2 Click the **Tester** icon (see Figure 56).

Figure 56 OTD Tester



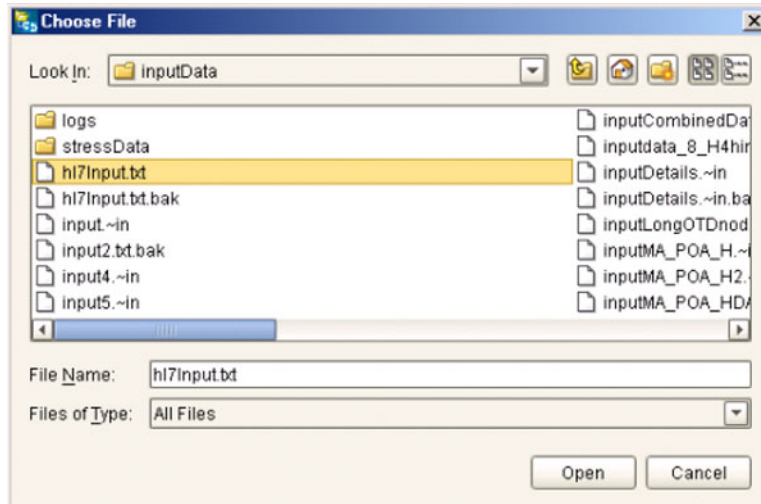
A test panel will appear below the OTD detail area of the editor. Note that there are three data display modes, selectable by tabs (see Figure 57). The Input tab is selected by default.

Figure 57 Test Panel Data Display



- 3 You can provide the input test data either by selecting a data file (see Figure 58), or by entering the data manually.

Figure 58 Select Data File



- 4 Click the **Run Tester** icon (green arrow) to test the selected OTD.
- 5 Verify the output by checking the values for each element for correctness (see Figure 59).

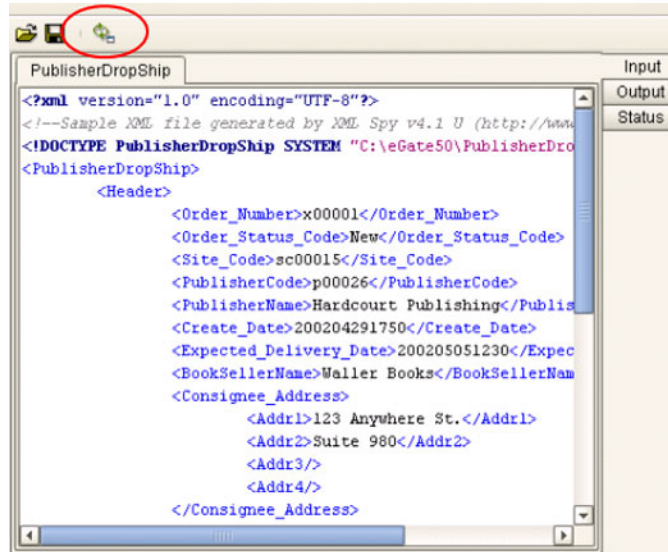
Figure 59 Object Elements and Values

Name	Value
☐ PublisherDropShip	
☐ header	
- name	""
- order_Number	"x00001"
- order_Status_Code	"New"
- site_Code	"sc00015"
- publisherCode	"p00026"
- publisherName	"Hardcourt Publi..."
- create_Date	"200204291750"
- expected_Delivery	"200205051230"
- bookSellerName	"Waller Books"
☐ consignee_Address	
- bom_type	""
- gl_entity	"GLN"
☐ terms	
☐ line_item	
- length	1
☐ [0]	
- value	"500"
- counter	"0"
- itemCode	"ISBN000139298"
- itemDescription	"King James Bib..."
- qty	"100"
- cost	"5.00"

- 6 You can save your input test data to a file for re-use by selecting the **Input** data display and clicking the **Save** icon.

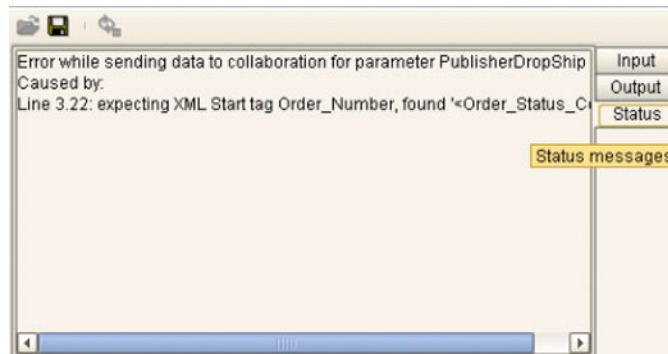
- 7 You can also change your test data in the Input data display, then re-test the OTD by clicking the **Refresh** icon (see Figure 60) to repopulate your OTD object elements with the new values.

Figure 60 Data Display: Refresh Icon



- 8 If there are errors in your input data, the **Status** data display is automatically invoked, showing the appropriate error messages (see Figure 61).

Figure 61 Status Data Display



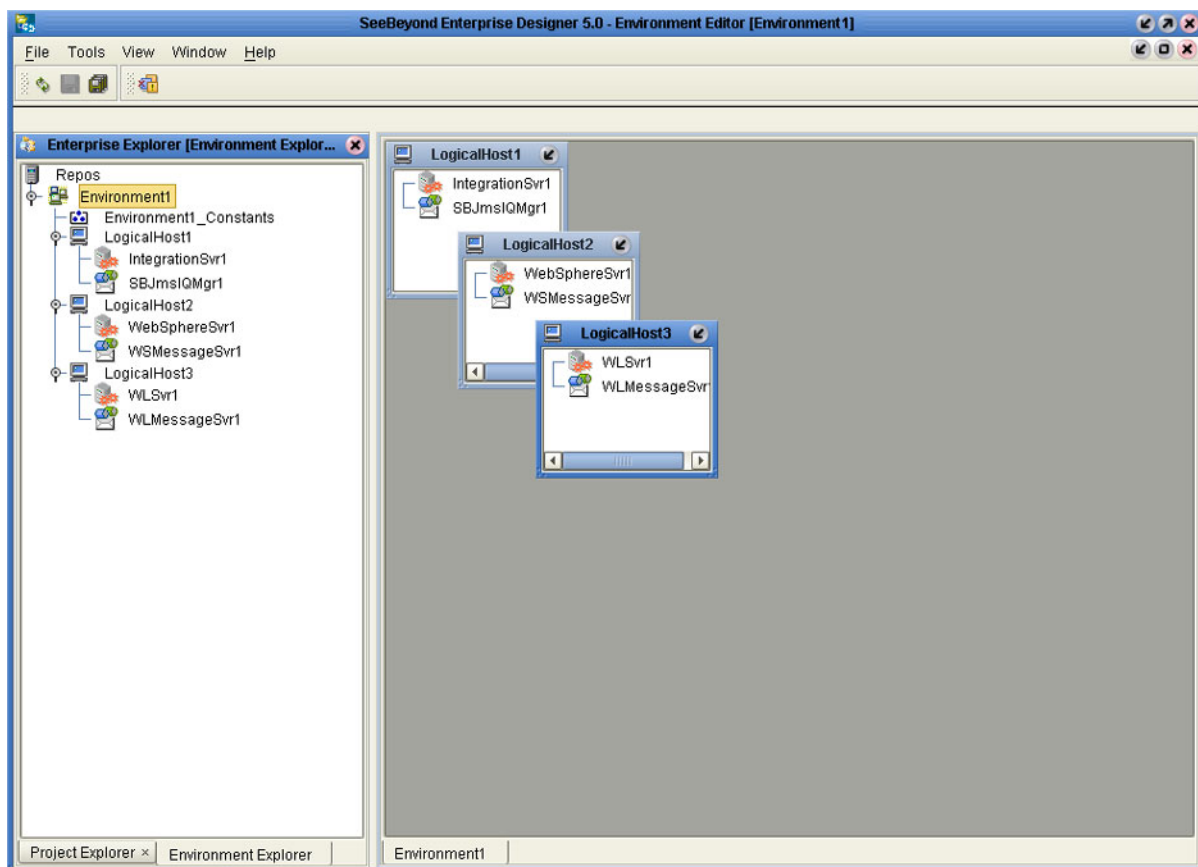
Environments and Project Deployment

This chapter describes the process of defining environments, creating deployment profiles, and activating the deployed projects.

6.1 Overview

Clicking on an Environment icon in the Environment Explorer invokes the Environment Editor, which provides a canvas in which you can create and customize an Environment (see Figure 62).

Figure 62 Environment Editor



Here you can see the various components (Logical Hosts, servers, and external systems) included in the selected Environment. New Environments are added through the use of the Repository context menu (see Figure 63). Components are added to the Environment by selecting options in the Environment and Logical Host context menus (see Figure 64 and Figure 65. See [Context Menus](#) on page 38 for additional information.

Figure 63 Repository Context Menu



Figure 64 Environment Context Menu

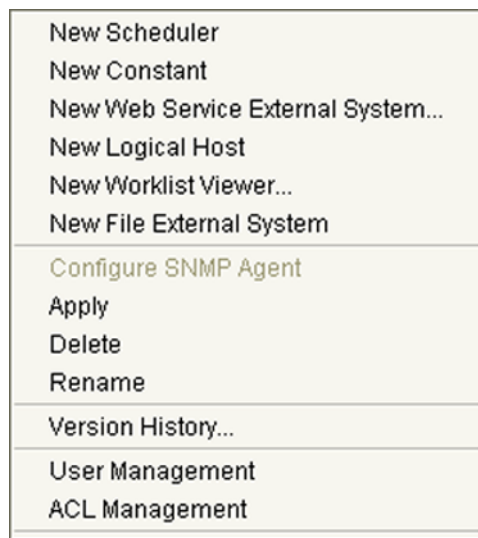
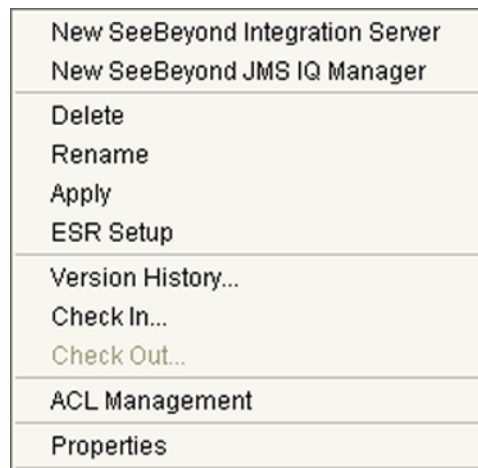


Figure 65 Logical Host Context Menu



6.2 Environmental Constants

You can define constants for a specific Environment. Environmental constants are name/value pairs that are visible across the Environment. When you create a constant you assign a permanent value to it, which cannot be overridden.

Selecting the **New Constant** option from the Environment context menu displays the Constants panel in the Environment Editor (see Figure 66).

Figure 66 Environmental Constants Editor

Name	Constant	Value	Category	Description
------	----------	-------	----------	-------------

Environment1: Add a New Constant

Name:

Category:

Description:

Is a Constant



Value Type:

Value:

OK

New constants are added using the bottom panel in the Constants editor. All constants defined for the specific Environment are listed in the top panel, along with their various properties.

Table 21 Environmental Constants Editor Icons

Icon	Name	Function
	Add a New Constant	Adds a new constant to the list.
	Delete a Highlighted Constant	Deletes the selected constant from the list.

6.3 Deployment Profiles

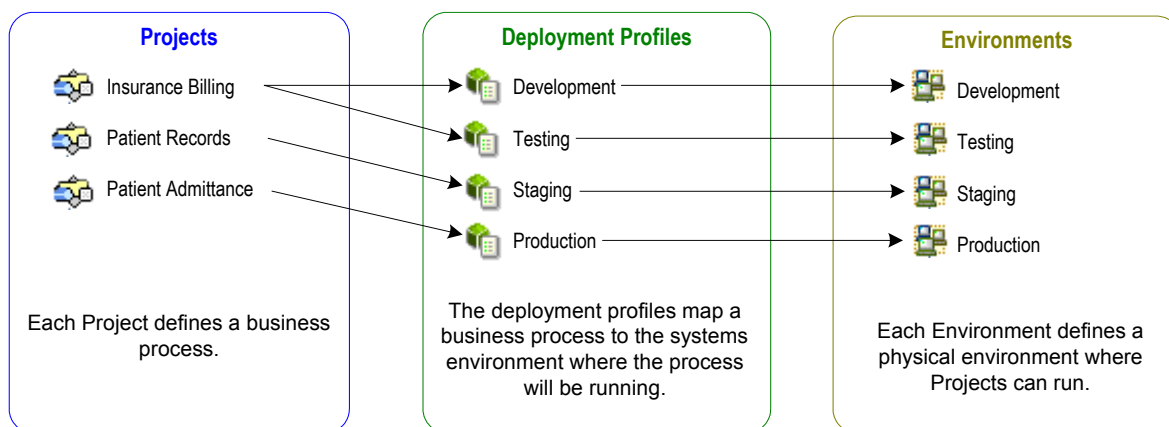
Deployment Profiles are specific instances of a Project in a particular Environment. A deployment profile contains information about the assignment of Services and Message Destinations to integration and message servers (JMS IQ Managers). It also contains version information for all relevant objects in the Project. The Enterprise Designer includes a Deployment Editor, which you can use to create and customize deployment profiles.

Note that:

- Each Project can have zero or more Deployment Profiles, but each of a Project’s active Deployment Profiles must be in a separate Environment.
- Each Environment can have zero or more Deployment Profiles assigned to it, but any given Environment can have only one Deployment Profile from a given Project.

Repeating Figure 3 from the [System Overview](#) on page 18:

Figure 67 eGate Integrator Implementation Model



6.4 Using the Deployment Editor

The Deployment Editor (see Figure 68) appears when you create a new Deployment Profile or click on an existing Deployment Profile icon in the Project Explorer tab.

Figure 68 Deployment Editor Window

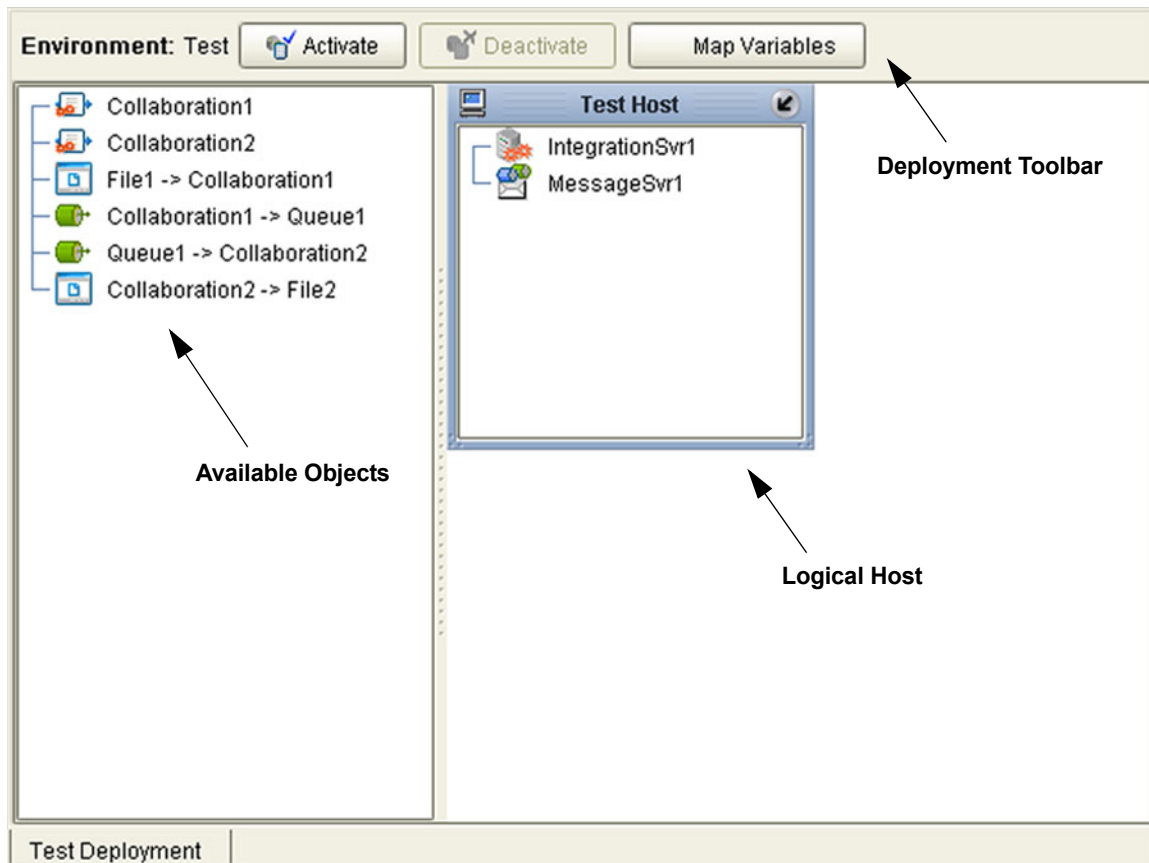




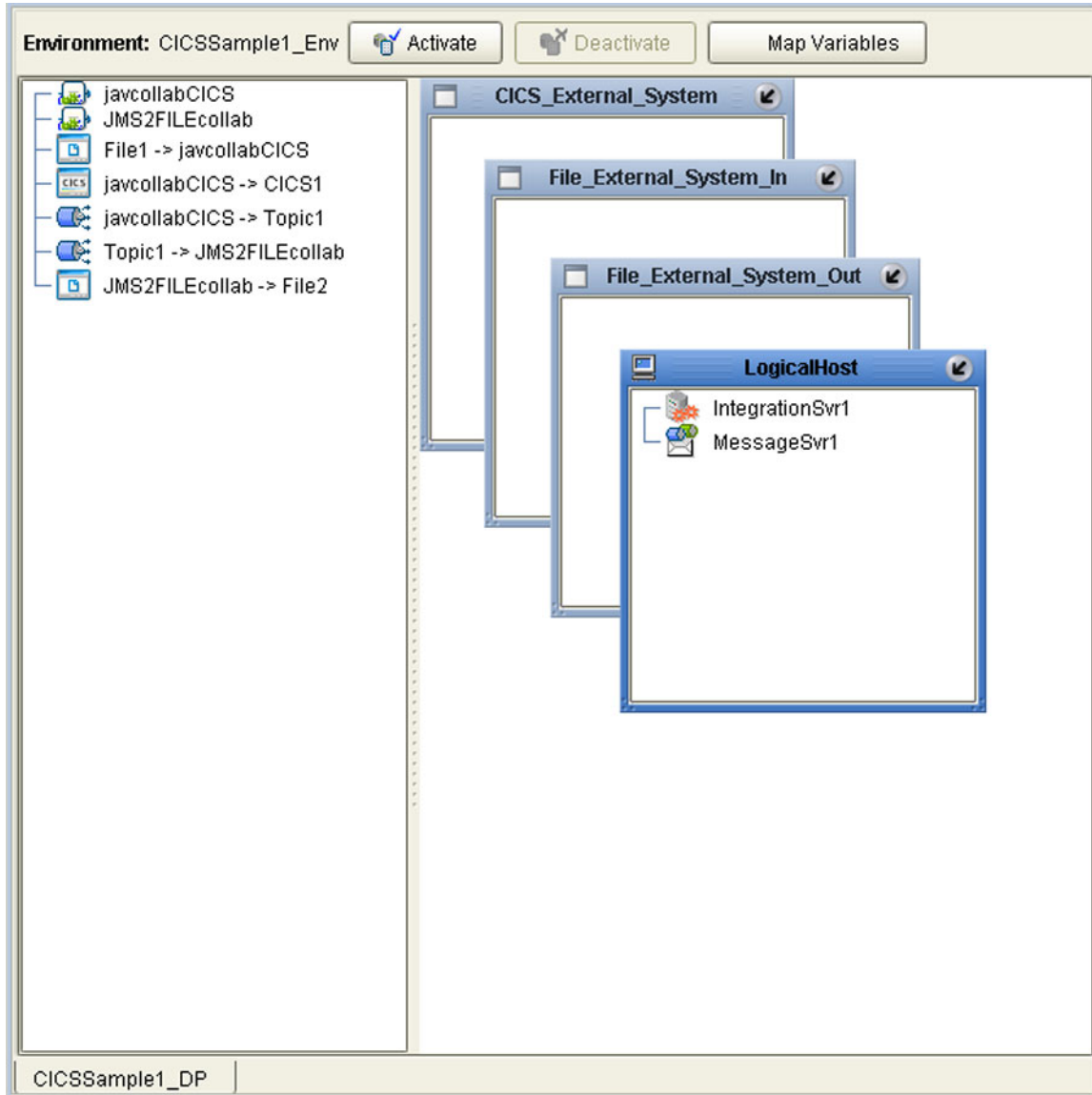
Table 22 Deployment Toolbar Buttons

Button	Function
 Activate	Starts the Project by creating an enterprise archive (EAR) file based on the Connectivity Map and linking this file with the SeeBeyond Integration Server. See Activating and Deactivating Deployments on page 96.
 Deactivate	Stops the Project by terminating the link between the EAR file and the SeeBeyond Integration Server, sets the Deployment Profile to <i>inactive</i> , and saves to the Repository.
Map Variables	Allows you to assign names and values to Project variables for the specific Deployment Profile. See Mapping Variables on page 98.

6.5 Creating a Deployment Profile

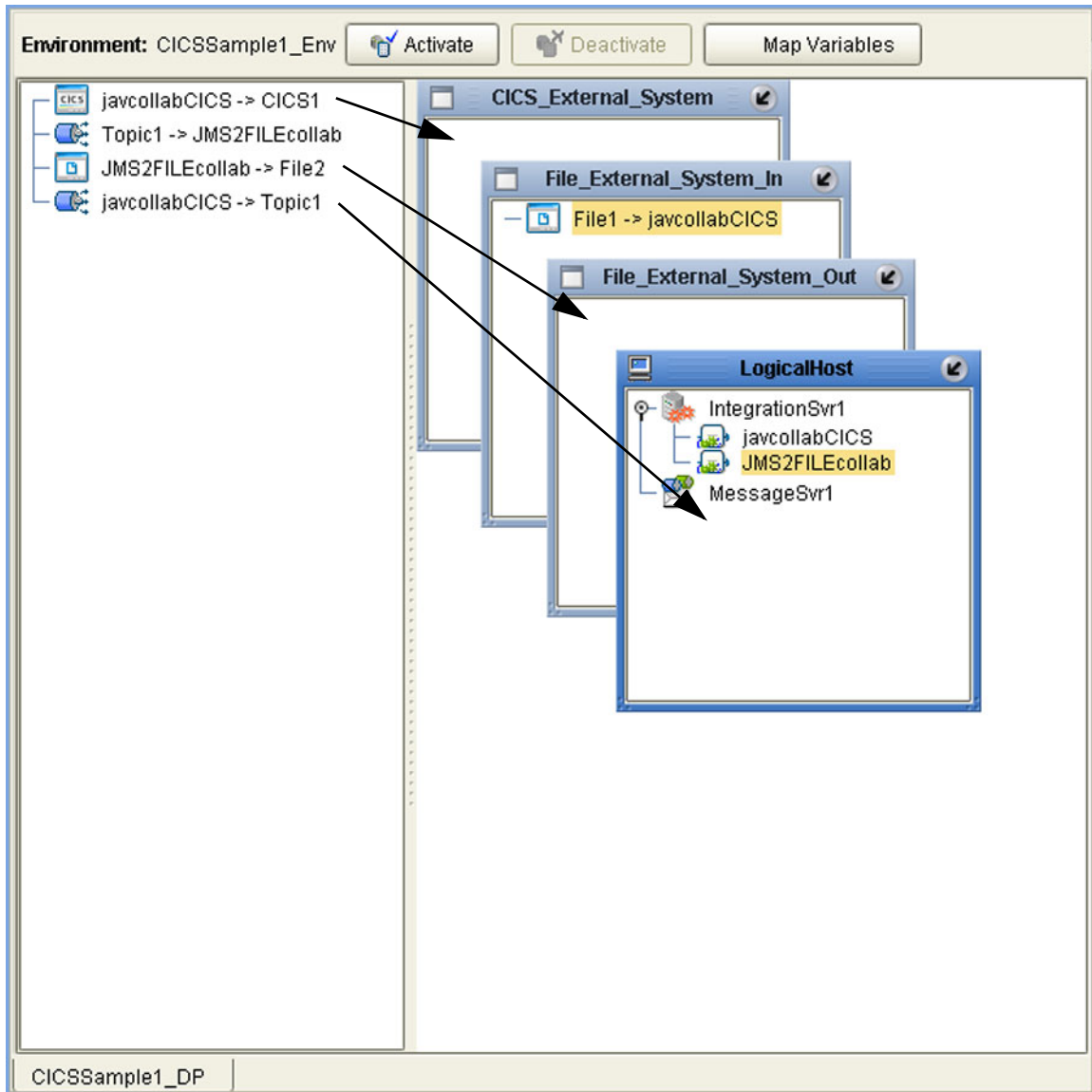
In the Environment Explorer, create an Environment and right-click on the Environment to display its context menu. From the menu, select the Environment components you need and name them appropriately. They will appear on the Deployment Editor canvas, as shown in Figure 69.

Figure 69 Example Deployment Profile (1)



Drag the Project components from the left panel and drop them into the appropriate Environment components in the right panel, as illustrated in Figure 70. As you do so, they will disappear from the left panel.

Figure 70 Example Deployment Profile (2)

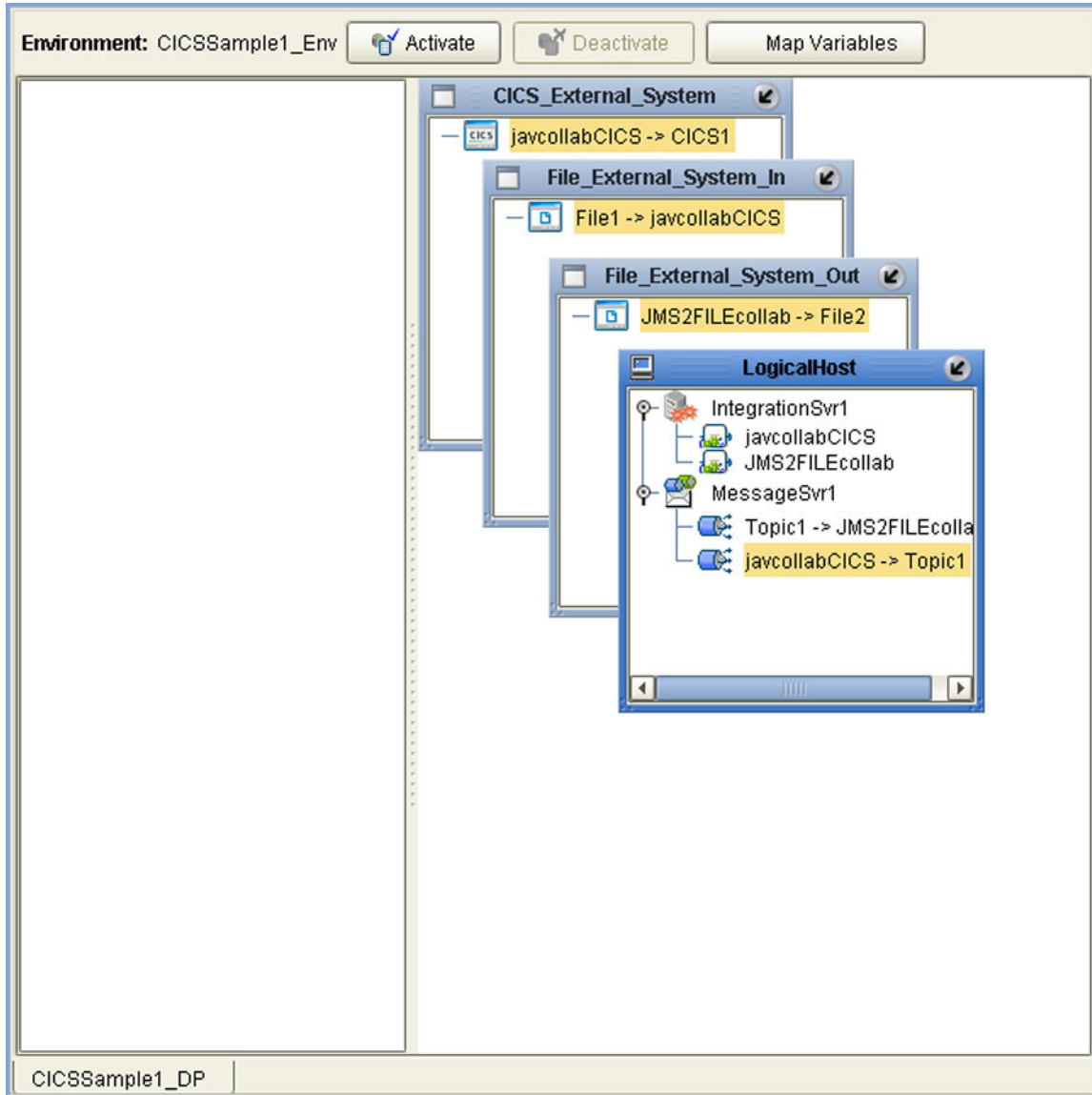


Note that:

- The eWay objects are placed into their appropriate External Systems.
- Collaboration objects are placed into the appropriate Integration Server on the appropriate Logical Host.
- Topic and queue objects are placed into the appropriate Message Server (JMS IQ Manager) on the appropriate Logical Host.

When the Environment components are fully populated, the left panel will be blank, as shown in Figure 71. The Deployment Profile is now ready to be saved and Activated.

Figure 71 Example Deployment Profile (3)



6.6 Activating and Deactivating Deployments

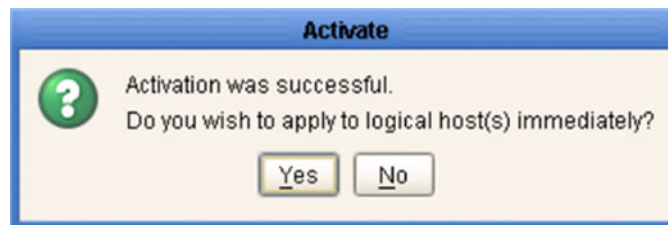
Using the Activate and Deactivate toolbar buttons, you have the option of immediately applying the changes to the Logical Host or deferring the changes to a later time. Activating the Deployment Profile without applying the changes checks the validity of the entire Deployment Profile.

Another advantage to activating the Deployment Profile without applying the changes comes into play when you have multiple Deployment Profiles to deploy at once. To save time, you can activate each of the Deployment Profiles without applying the changes. Then when you do apply all of the changes to the Logical Host in one batch.

To activate a Deployment Profile

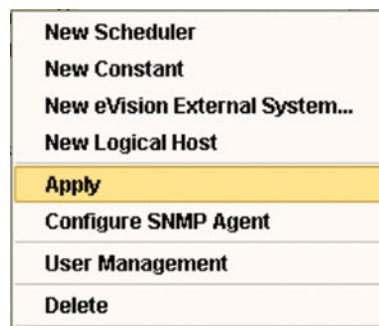
- 1 In the Deployment Profile, select the Deployment you wish to activate.
- 2 Click the **Activate** button. The following message appears:

Figure 72 Activate Dialog Box



- 3 Answer the question following these criteria:
 - If the Logical Host is running, and you wish to apply the changes immediately, click **Yes**.
 - If the Logical Host has not yet been bootstrapped, or you wish to apply the changes at a later time, click **No**. To apply the changes later, right-click the Logical Host and select **Apply** from the menu (see Figure 73). This will apply all of the changes for that Logical Host.

Figure 73 Logical Host Context Menu - Apply

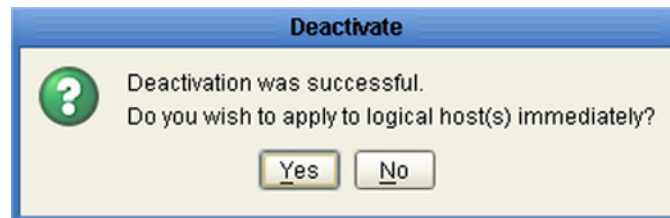


Note: The **Apply** action assumes that the Logic Host is running, since it invokes a trigger to the Logical Host causing it to download the latest settings from the Repository and deploy those settings to all components on the Logical Host.

To deactivate a Deployment Profile

- 1 In the Deployment Profile, select the Deployment you wish to deactivate.
- 2 Click the **Deactivate** button. The following message appears:

Figure 74 Activate Dialog Box



- 3 Answer the question following these criteria:
 - If the Logical Host is running, and you wish to apply the changes immediately, click **Yes**.
- 4 If the Logical Host has not yet been bootstrapped, or you wish to apply the changes at a later time, click **No**. To apply the changes later, right-click the Logical Host and select **Apply** from the menu (see Figure 73). This will apply all of the changes for that Logical Host. See the **Note** following the activation procedure.

Note: In Windows and NFS, application working directories cannot be deleted during deactivation. This is because *.jar* files in these directories have been added to a *ClassLoader* and the JVM maintains *locks/handles?* on any such files. At the subsequent startup of the Integration Server, leftover work directories in the repository/application directory are deleted.

6.7 Mapping Variables

Project variables function as placeholders, having values that are determined when you create a specific Deployment Profile. These values can be literals or Environmental constants. Clicking the **Map Variable** button displays the Deployment Profile Mappings panel, where you can assign names (see Figure 75) and values (see Figure 76).

Figure 75 Deployment Profile Mappings

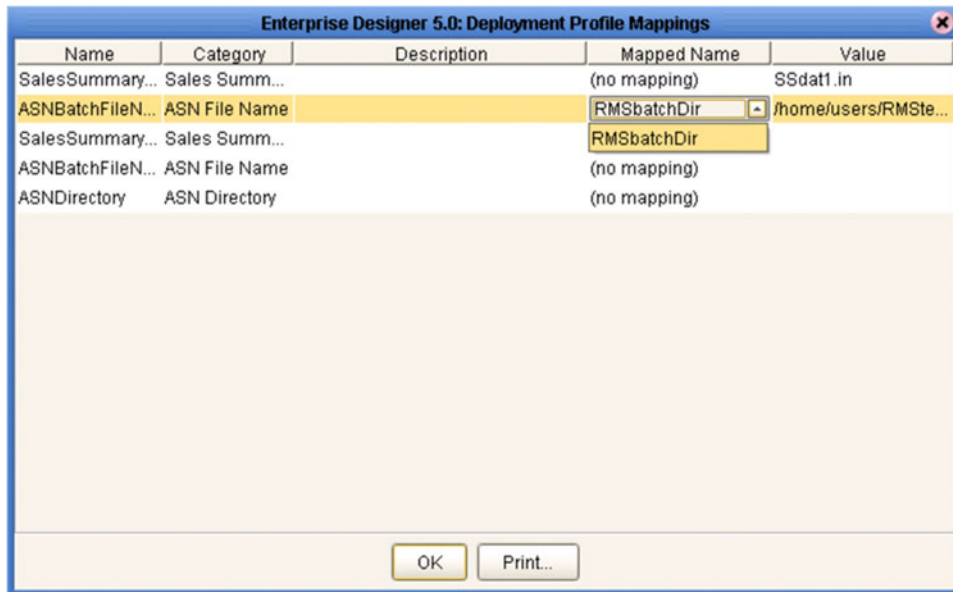
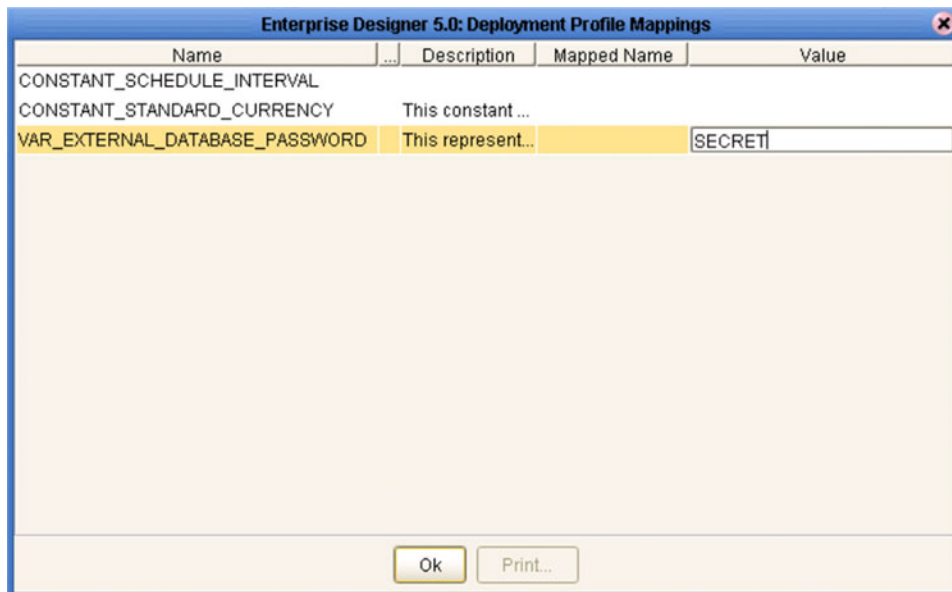


Figure 76 Project Variable Value Entry



6.8 Deploying Projects to Third-Party Servers

SeeBeyond's eGate Integrator allows you to develop Projects using Enterprise Designer and deploy them to a BEA WebLogic or IBM WebSphere environment. The SAR files for these third-party products must be installed prior to deployment, as described in the *eGate Integrator JMS Reference Guide*.

Important: The file `log4j.jar` must be added to the classpath for both WebLogic and WebSphere. This file can be obtained from the following URL:

<http://jakarta.apache.org/log4j/docs/>

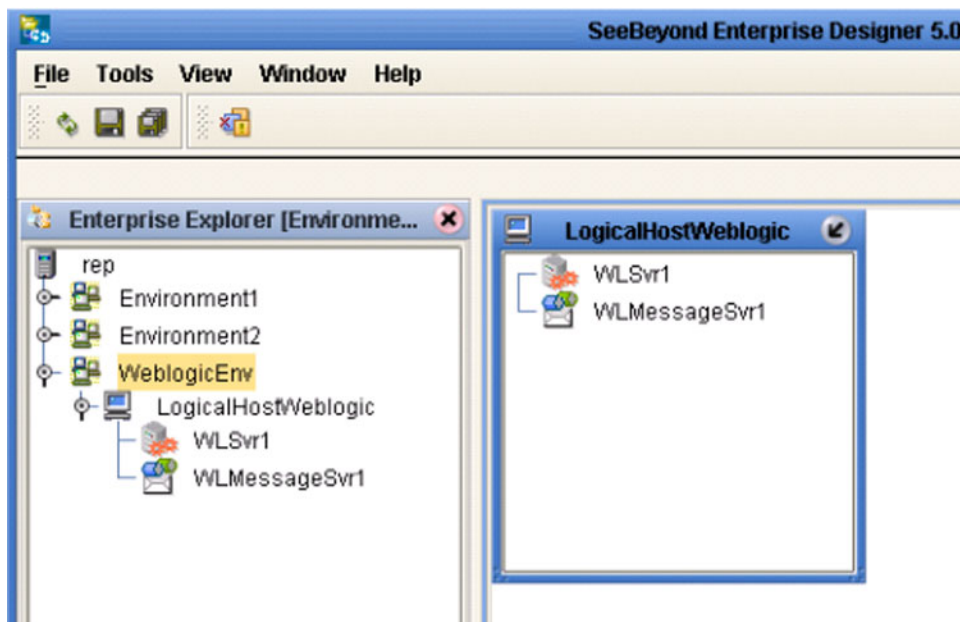
Note: The initial capability provided by eGate Integrator 5.0 is restricted to topic-to-topic Collaborations and JMS connections.

6.8.1 BEA WebLogic

To deploy an eGate Project to a BEA WebLogic environment

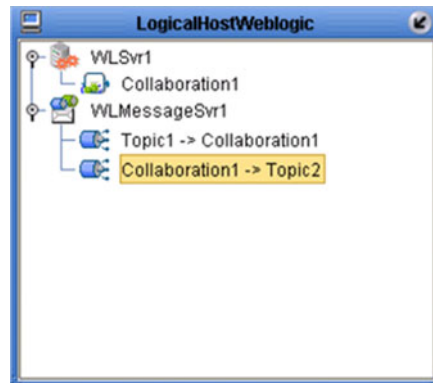
- 1 Create the following components in Enterprise Designer (see Figure 77):
 - A A new environment
 - B A Logical Host
 - C A WebLogic Integration Server
 - D A WebLogic JMS Message Server

Figure 77 WebLogic Deployment (1)



- 2 Create a new Deployment Profile to bind the Connectivity Map to the new WebLogic environment (see Figure 78).
 - A Drag the two topics and drop onto the WebLogic message server.
 - B Drag the Collaboration and drop onto the WebLogic integration server.

Figure 78 WebLogic Deployment (2)



- 3 Activate the Deployment Profile.

Activating the Deployment Profile creates an Environment Archive (EAR) file, which contains all files necessary to create and run an application in WebLogic. This file can be found in the following location (where **eGate50** is your eGate root directory):

```
eGate50\repository\data\files\WLEnvironmentName\  
ProjectName_DeploymentProfileName.ear
```

Note: The remainder of this procedure is performed in the WebLogic user interface, and is only outlined here. Please refer to your BEA WebLogic documentation for current information regarding interface layout and deployment details.

- 4 Start the BEA WebLogic server.
- 5 Navigate to **Server Administration Console > Deployments > Applications**.
- 6 Perform the following steps:
 - A Add a new JMS Connection Factory.
 - B Enter a JNDI name for the JMS Connection Factory:

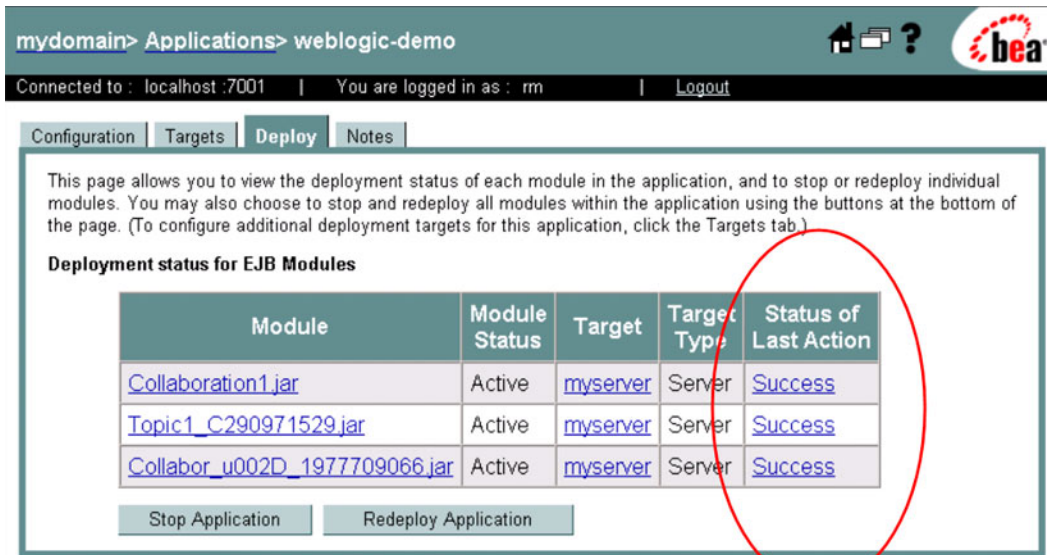
```
jms/connectionfactory/xa-topic/  
LogicalHostName_MessageServerName
```

For example, the default name would be:

```
jms/connectionfactory/xa-topic/LogicalHost1_WLMessageSvr1
```
 - C Verify that the WebLogic JMS Server Destination names for topics match those in eGate.
 - D Select **Deploy a new Application**.
 - E Upload and install the EAR file described in step 3.

- F Select the EAR file you just installed as the archive for the new application.
- G Enter a name for the new application.
- H Click **Deploy**.
- I Verify the success of the deployment (see Figure 79 , which shows a WebLogic 8.1 example).

Figure 79 WebLogic Deployment Verification

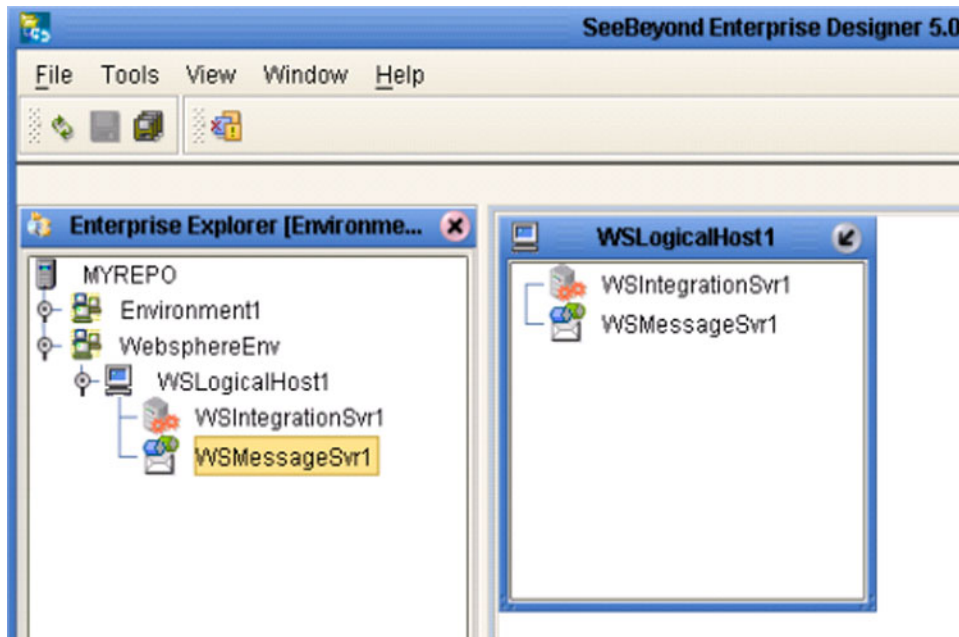


6.8.2 IBM WebSphere

To deploy an eGate Project to an IBM WebSphere environment

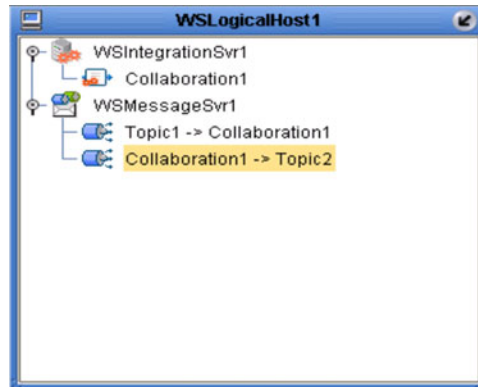
- 1 Create the following components in Enterprise Designer (see Figure 80):
 - A A new environment
 - B A Logical Host
 - C A WebSphere Integration Server
 - D A WebSphere JMS Message Server

Figure 80 WebSphere Deployment (1)



- 2 Create a new Deployment Profile to bind the Connectivity Map to the new WebSphere environment (see Figure 81).
 - A Drag the two topics and drop onto the WebSphere message server.
 - B Drag the Collaboration and drop onto the WebSphere integration server.

Figure 81 WebSphere Deployment (2)



- 3 Activate the Deployment Profile.

The activated Deployment Profile creates an Environment Archive (EAR) file, which contains all files necessary to create and run an application in WebSphere. This file can be found in the following location (where **eGate50** is your eGate root directory):

```
eGate50\repository\data\files\WSEnvironmentName\  
ProjectName_DeploymentProfileName.ear
```

Note: *The remainder of this procedure is performed in the WebSphere user interface, and is only outlined here. Please refer to your IBM WebSphere documentation for current information regarding interface layout and deployment details.*

- 4 Start the IBM WebSphere server.
- 5 From the Administrative Console, navigate to **Servers > Application Servers > server_name > Message Listener Service > Listener Ports**.
- 6 Add a new Listener port.
- 7 Enter a Connection Factory JNDI name for the new port:

```
jms/connectionfactory/xa-topic/  
LogicalHostName_MessageServerName
```

For example, the default name would be:

```
jms/connectionfactory/xa-topic/LogicalHost1_WSMMessageSvr1
```

This binds the JNDI name with the WebSphere Message Server Listener port.

- 8 From the Administrative Console, navigate to **Applications > Enterprise Applications > Install New Application**.
- 9 In *Preparing for the application installation*:
 - A Enter the path for the EAR file described in step 3 and click **Next**.
 - B Select **Generate Default Bindings** and click **Next**.

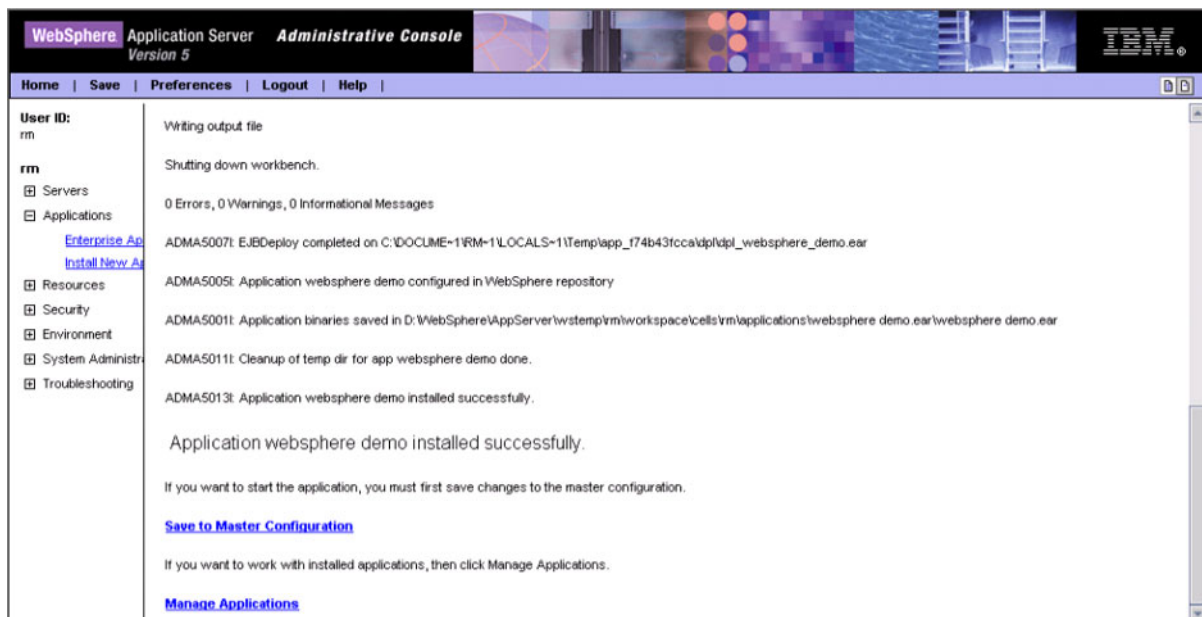
- 10 In *Step 1, Provide options ...*:
 - A Check **Deploy EJBs**.
 - B Enter the application name.
 - C Click **Next**.
- 11 In *Step 2, Provide options ...*, click **Next**.
- 12 In *Step 3, Provide Listener Ports ...*, accept the default value and click **Next**.

Note: The Listener port number should match the port number entered in step 6.

Note:

- 13 In *Step 4, Provide JNDI Names ...*, accept the default value and click **Next**.
- 14 In *Step 5, Provide EJB references ...*, accept the default value and click **Next**.
- 15 In *Step 6, Map resource references ...*, enter the JNDI name from step 7, and click **Next**.
- 16 In *Step 7, Map modules ...*, check all modules and click **Next**.
- 17 In *Step 8, (protection levels)*, check all modules and click **Next**.
- 18 In *Step 9, Summary*, click **Finish**.
- 19 Verify the success of the deployment (see Figure 82 , which shows a WebSphere 5 example).

Figure 82 WebSphere Deployment Verification



Logical Hosts

This chapter describes the procedures for configuring and running a Logical Host.

7.1 Overview

A Logical Host is an instance of the eGate runtime environment that is installed on a host hardware platform. A Logical Host can be a member of only one Environment, but each Environment can contain multiple Logical Hosts. When multiple Logical Hosts reside on a single hardware platform, you must configure the base port numbers (see [Configuring a Logical Host](#) on page 107).

7.1.1 Integration Servers

The Logical Host contains one or more Integration Servers, which are the engines that run eGate Collaborations and eWays. See [Integration Servers](#) on page 106.

7.1.2 Message Servers

The Logical Host contains one or more Message Servers, which manage JMS topics (publish-and-subscribe messaging) and queues (point-to-point messaging). eGate Integrator includes the SeeBeyond JMS IQ Manager as its Java Messaging Service (JMS) implementation. The JMS IQ Manager conforms to the Java Message specification 1.0.2b, and supports both topic (publish-and-subscribe) and queue (point-to-point) messaging styles.

eGate also includes support for HP NonStop JMS for eGate implementations on HP NonStop Server platforms. Third-party integration servers such as BEA WebLogic and IBM WebSphere incorporate their own Message Servers. For more information on the JMS IQ Manager, and deploying Project components to third-part message servers, see the *eGate Integrator JMS Reference Guide*.

7.1.3 Management Agent

The Management Agent is the master service of the Logical host. The service starts the other services on the Logical Host as part of the bootstrap process. The Management Agent also communicates with the Enterprise Manager via JMX (Java Management Extensions) to report the status of the JMS IQ Managers and Integration Servers.

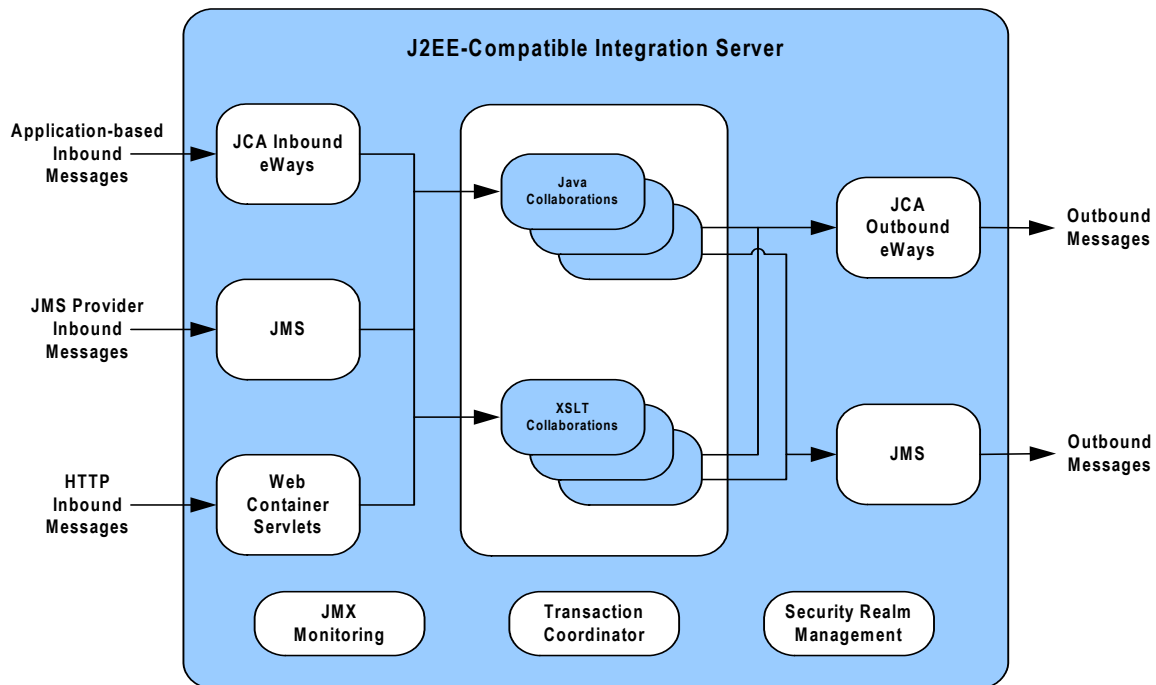
7.1.4 Bootstrap

The bootstrap is a process that launches the Management Agent and subsequent services. Each Logical Host has a separate bootstrap process. The process is started from a batch file (bootstrap/bin/bootstrap.bat). It finds the Repository via command-line parameters or from the configuration file (bootstrap/config/logical-host.properties). See [Starting the Logical Host](#) on page 112.

7.2 Integration Servers

The Integration Server is the engine that runs eGate Collaborations for processing business logic and eWays that communicate with external applications. It provides services for security, transactions, business rules execution, and connectivity management. The SeeBeyond Integration Server is based on Java 2 Enterprise Edition (J2EE). eGate Integrator also runs on third-party integration servers such as BEA WebLogic and IBM WebSphere (see [Deploying Projects to Third-Party Servers](#) on page 99).

Figure 83 Integration Server (J2EE Compatible)



The Integration Server runs the JCA eWays and Collaborations. For HTTP/HTTPS, rather than using a JCA adapter, the Integration Server uses servlets running in the Web container to provide J2EE HTTP facilities.

7.3 Configuring a Logical Host

7.3.1 Logical Host Startup Parameters

The startup command syntax is as follows:

```
bootstrap arguments
```

Table 23 Command Arguments for bootstrap

Parameter	Description	Req/Opt
-d <i>debug</i>	Overrides bootstrap sequence. Displays all cached (default) argument values	Optional
-e <i>environment name</i>	The name of the Environment to which this Logical Host belongs.	Required (first time only)
-h <i>help</i>	Overrides bootstrap sequence. Displays the usage report.	Optional
-i <i>id</i>	The user ID used for accessing the Repository. Note that the user ID is the same as the username, and that the Administrator can set up more than one user ID.	Required (first time only)
-l <i>logicalhost name</i>	The name of this Logical Host.	Required (first time only)
-n <i>physical host name</i>	The name of this Physical Host.	Required (first time only)
-p <i>password</i>	The password used for accessing the Repository.	Required (first time only)
-r <i>repository URL</i>	The root URL for the Repository containing the Logical Host data.	Required (first time only)

Note: *Required (first time only)* indicates that the argument is required the first time you start the Repository. You do not need to use it again unless you change the name of the Logical Host.

7.3.2 Modifying the Logical Host Startup Configuration File

Instead of setting the arguments when you start the Logical Host at the command prompt, you can preset them in the **logical-host.properties** file. Although set, the arguments can be overridden by starting the Logical Host as described in [To start the Logical Host](#) on page 114.

To modify the startup configuration file

- 1 Navigate to `c:\eGate50\logicalhost\bootstrap\config\logical-host.properties`.
- 2 Use a text editor (such as Windows WordPad) to open the file.

Figure 84 Example logical-host.properties File

```
#####
#
#                               Logical Host Properties
#
#####

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

#####
# repository.url: (USER MODIFIABLE)
#   Specifies the remote URL for connecting to the repository.
#   Takes the form:
#       http://<repository-server-hostname>:<port>/
#       <repository-name>
#   For example:
#       http://localhost:10000/myRep
#####
repository.url=

#####
# repository.username: (USER MODIFIABLE)
#   Username for connecting to the repository.
#####
repository.username=

#####
# repository.password: (USER MODIFIABLE)
#   Plain text form of password used for connecting to the
#   repository. Any value provided here will be cleared out
#   by the system and written in encrypted form to the
#   repository.password.encrypted field.
#####
repository.password=

#####
# repository.password.encrypted:
#   Encrypted form of the repository password. NOTE: This value
#   is generated by the system, so it is improper to edit this
#   field manually.
#####
repository.password.encrypted=

#####
# physical.host.name: (USER CONFIGURABLE)
#   Specifies the physical host on which this logical host is
#   running. The host name should include the domain name.
#   Example: host.company.com
#####
physical.host.name=

#####
# logical.host.environment.name: (USER MODIFIABLE)
#   Specifies the name of the environment containing the
#   current logical host.
#####
logical.host.environment.name=

#####
# logical.host.name: (USER MODIFIABLE)
#   Specifies the name of the current logical host.
#####
logical.host.name=
```

```
#####
# logical.host.root.dir:
#       Specifies the root directory of a logical host
#       installation.
#####
logical.host.root.dir=

#####
# os.type:
#       Specifies the OS type of the machine on which logical host
#       is going to run
#####
os.type=

# *** THE FOLLOWING PROPERTIES ARE PRIVATE; MODIFY AT YOUR OWN RISK ***

# bootstrap
managementagent.jar.path=lib/managementagent.jar

# deployment manager
deployment.manager.jar.relative.path=deploymentmanager.jar

# management agent
managementagent.config.file=./config/ManagementAgent-config.xml
managementagent.command.line.windows=cmd.exe /c ..\jre\bin\java.exe -Xrs -
Dlogical.host.properties.file=..\bootstrap\config\logical-host.properties -classpath
..\config;..\lib\managementagent.jar;..\lib\com.stc.hnsems.jar;..\lib\jta.jar;..\lib
\jms.jar;..\lib\jta.jar;..\lib\concurrent.jar;..\lib\log4j.jar;..\lib\jargs.jar;..\
\lib\mx4j.jar;..\lib\mx4j-
tools.jar;..\lib\stcjs.jar;..\lib\stcrepository.jar;..\lib\stccore.jar;..\lib\stcr
epositoryclient.jar;..\lib\stcrepositoryserver.jar;..\lib\velocity.jar;..\lib\ejb.ja
r;..\lib\commons-
collections.jar;..\stcis\lib\com.stc.isbootstrap.jar;..\stcis\lib\gnu-regexp-
1.1.4.jar;..\lib\com.stc.isapi.jar;..\stcis\lib\com.stc.isimpl.jar;..\lib\com.stc.
compatimpl.jar;..\lib\com.stc.eventmanagementapi.jar;..\lib\xmlrpc.jar;..\lib\smscli
ent.jar;..\lib\com.stc.stcutil4Jimpl.jar;..\lib\xercesImpl.jar;..\lib\xalan.jar;..\l
ib\xml-apis.jar;..\lib\stcqueueviewer.jar;..\lib\stcjournaller.jar;..\lib\jakarta-
regexp-1.2.jar;;..\lib\commons-codec-1.1.1-dev.jar;
com.stc.processmanager.ma.ManagementAgent

managementagent.command.line.unix=sh ../bootstrap/bin/magent.sh

managementagent.shutdown.windows=cmd.exe /c ..\jre\bin\java.exe -
Djava.ext.dirs=..\lib com.stc.sms.mbeans.ManagementAgentControl -f
..\config\ManagementAgent-config.xml -c shutdown

managementagent.shutdown.unix=sh ../jre/bin/java -Djava.ext.dirs=./lib
com.stc.sms.mbeans.ManagementAgentControl -f ../config/ManagementAgent-config.xml -c
shutdown

# misc
repository.relative.url=data/repository/SBYN00.properties
log4j.config.path=bootstrap/config/log4j.xml
jre.zip.name=jre.zip
```

- 3 In the top part of the **logical-host.properties** file, locate the following properties (they may appear in a different order than that shown):

```
logical.host.environment.name=
logical.host.name=
logical.host.root.dir=
os.type=
physical.host.name=
repository.url=
repository.username=
repository.password=
repository.password.encrypted=
```

- 4 Set the properties in your **logical-host.properties** file. See the following table for descriptions of the properties' values:

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

Table 24 Logical Host Properties file

Property	Description
logical.host.environment.name	Specifies the name of the Logical Host environment deployment, for example, EnvironmentDeployment1 .
logical.host.name	Specifies the name for the Logical Host, for example, logicalhost1 .
logical.host.root.dir	Specifies the full path of the Logical Host directory, for example, c:\eGate50\logicalhost .
os.type	Specifies the operating system type under which logical host is going to run.
physical.host.name	Specifies the physical host on which this logical host is running. The host name should include the domain name, for example: host.company.com .
repository.url	Specifies the name of the Repository URL, for example http://hostname:port/repositoryname where: <ul style="list-style-type: none"> ▪ hostname is the physical name of the computer on which the Repository resides. For example, localhost. ▪ port is the port number that the Repository uses to receive requests. For example, 9999. ▪ repositoryname is the name you specified for the Repository. For example, Repository1.
repository.username	Specifies the user name you are using to access the Repository, for example, Administrator .
repository.password	Specifies the password you are using to access the Repository, for example, myPassword . When you launch the bootstrap process, this password is encrypted and written to the repository.password.encrypted property. After the encrypted password has been written, this repository.password value is removed from the logical-host.properties file.
repository.password.encrypted	This property is automatically updated based on changes made to the repository.password property. <i>Do not enter anything for this property or modify its contents.</i>

- 5 Save the **logical-host.properties** file.

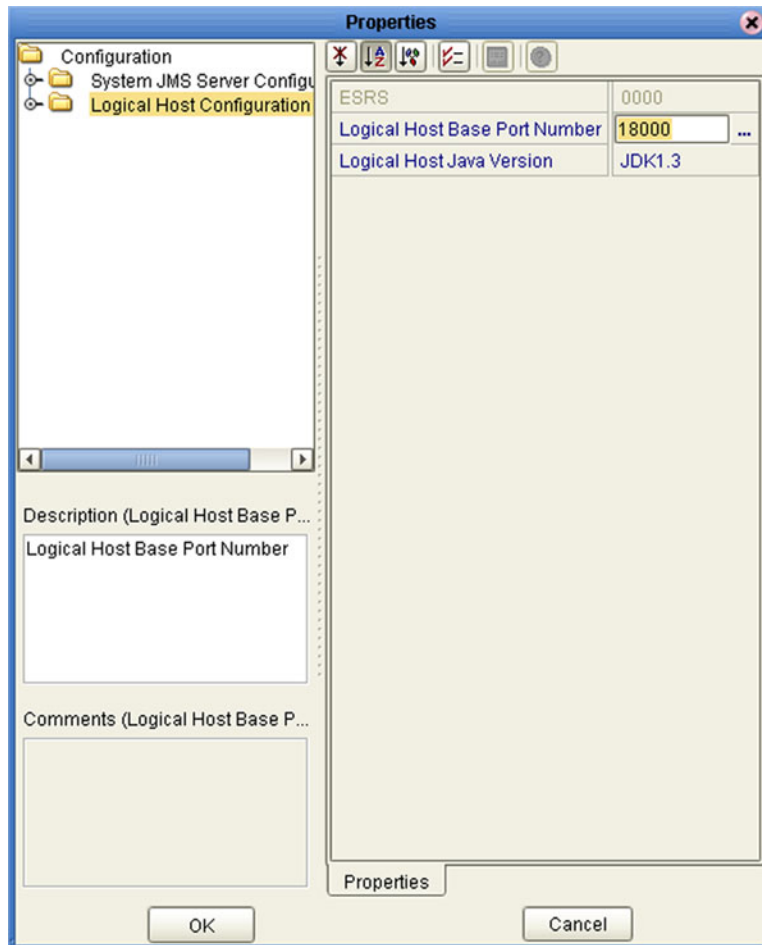
To start the Logical Host after setting the configuration file arguments

- 1 Navigate to **c:\eGate50\logicalhost\bin**.
- 2 Double-click **bootstrap.bat**.

7.3.3 Configuring the Base Port Number

Port numbers assigned to the Logical Host and its components (Integration Server, JMS IQ Manager, etc.) can be viewed and modified in the Enterprise Explorer by selecting **Properties** from the component's context menu (see Figure 85).

Figure 85 Logical Host Properties Dialog Box



If multiple Logical Hosts reside in the same Environment, you must ensure that each Logical Host has a different base port number to avoid conflicts. This base port number is propagated throughout the Logical Host, so that the various components are automatically given successive port numbers following that assigned to the Logical Host itself.

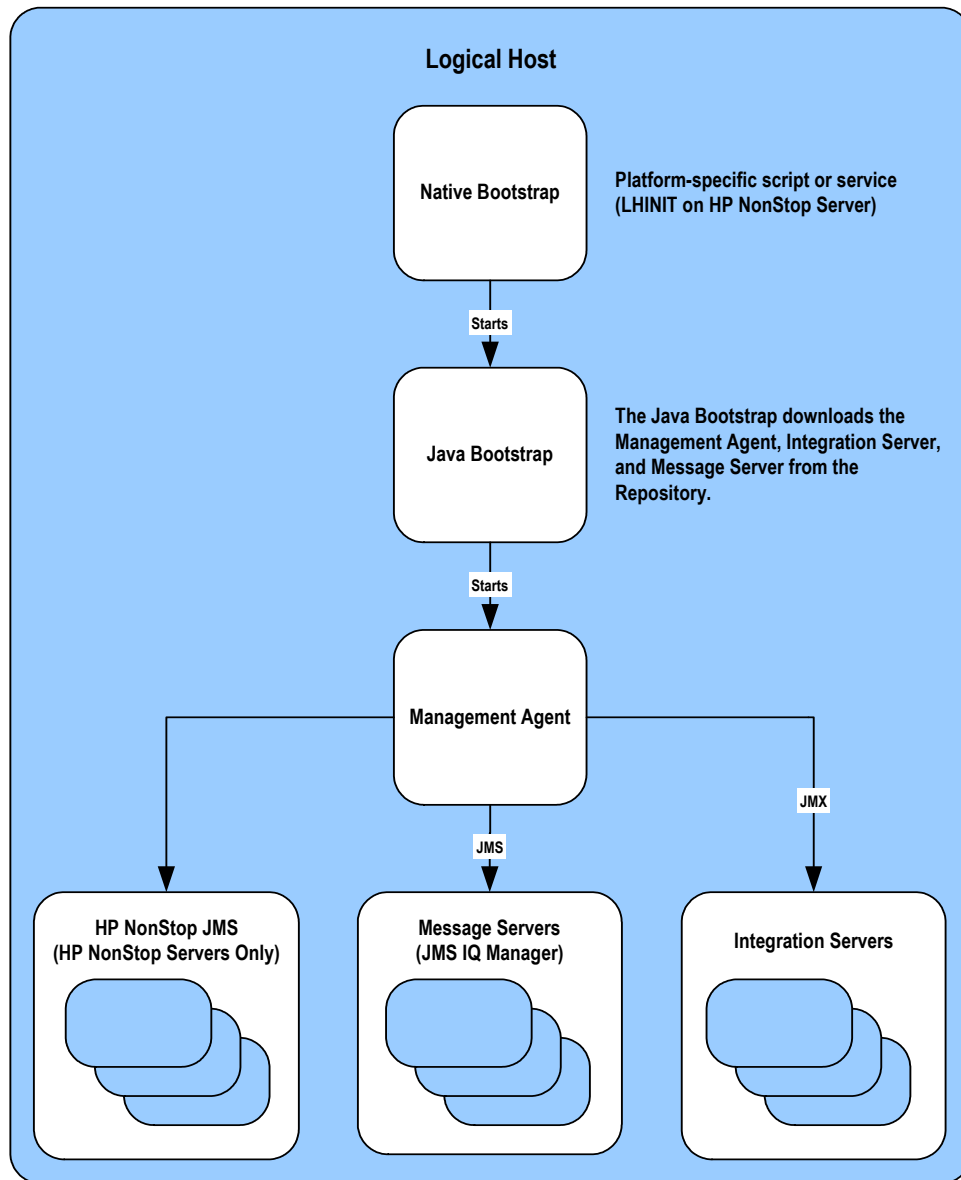
The number of port numbers used in a Logical Host varies according to the specific implementation, so when assigning new base port numbers you need to skip successive numbers by an adequate amount. The default base port number is 18000, so base port numbers of 19000, 20000, and so on are recommended.

If you need to assign a specific port number to a particular Logical Host component, you are allowed to do so. In this case, the automatic numbering process will skip the component port number you have assigned manually. Just be sure this port number is not used elsewhere.

7.4 Starting the Logical Host

The Logical Host can be started manually by using the bootstrap batch script specific to the platform where it is running (see Figure 86). Additionally, the Logical Host can be started as a service on Windows platforms only.

Figure 86 Logical Hosts



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

7.4.1 Starting the Logical Host as a Windows Service

Installing the Logical Host as a Windows service configures the Logical Host to automatically start up at system startup. This makes it possible for the Logical Host to automatically restart after an abnormal system shutdown.

Starting the Logical Host as a service requires you to have configured the **logical-host.properties** file. For instructions on configuring the **logical-host.properties** file, see [“Modifying the Logical Host Startup Configuration File” on page 108](#).

You must have Administrator rights to the local Windows machine in order to configure the Logical Host to start as a service. The installation script writes to the Windows Registry; this cannot be done without Administrator rights.

To start the Logical Host as a Windows Service

- 1 Use Windows Explorer to navigate to the Logical Host’s **bootstrap\bin** directory. For example, `c:\eGate50\LogicalHost\bootstrap\bin`.
- 2 Double-click **install-as-nt-service.bat**.

This runs the script that configures the Windows service to automatically run the Logical Host at system startup. The only parameter is the name of your Logical Host Service; the default is “ICAN 5.0 Logical Host” (see Figure 87).

Figure 87 Install as Service Script

```
C:\ican50\logicalhost\bootstrap\bin>install-as-nt-service.bat

-----
--- eGate 5.0 Management Agent Install Bootstrap NT Service ---
--- Copyright (c) 2003, SeeBeyond Technology Corporation, ---
--- All Rights Reserved ---
-----

To start the service use: net start "ICAN 5.0 LogicalHost"
To stop the service use: net stop "ICAN 5.0 LogicalHost"
C:\ican50\logicalhost\bootstrap\bin\..\..
C:\ican50\logicalhost\bootstrap\bin\..\..
C:\ican50\logicalhost\bootstrap\bin\..\..\bootstrap\lib
C:\ican50\logicalhost\bootstrap\bin\..\..\bootstrap\config
The service was successfully installed.
```

- 3 Verify the installation by opening the Windows Services facility and searching for the Logical Host name (see Figure 88). Note that by default, the service is listed as *Automatic*—it will not be running, however, until you click **Start** or reboot the computer.

Figure 88 Windows Services List

Hummingbird Ineco	One Hummi...		Manual	LocalSystem
Hummingbird Jconfi...	Jconfig Da...		Manual	LocalSystem
ICAN 5.0 LogicalHost			Automatic	LocalSystem
IIS Admin Service	Allows adm...	Started	Automatic	LocalSystem
Indexing Service		Started	Automatic	LocalSystem
Informatica			Manual	STC\ivanh

To remove the Logical Host Windows Service

- 1 Use Windows Explorer to navigate to the Logical Host's **bootstrap\bin** directory. For example, **c:\eGate50\LogicalHost\bootstrap\bin**.
- 2 Double-click **uninstall-nt-service.bat**.

This runs the script that removes the Windows service that automatically starts the Logical Host at system startup (see Figure 89).

Figure 89 Uninstall as Service Script

```
C:\ican50\logicalhost\bootstrap\bin>install-as-nt-service.bat

-----
eGate 5.0 Management Agent Install Bootstrap NT Service
Copyright (c) 2003, SeeBeyond Technology Corporation,
All Rights Reserved
-----

To start the service use: net start "ICAN 5.0 LogicalHost"
To stop the service use: net stop "ICAN 5.0 LogicalHost"
C:\ican50\logicalhost\bootstrap\bin\..\..
C:\ican50\logicalhost\bootstrap\bin\..\..
C:\ican50\logicalhost\bootstrap\bin\..\..\bootstrap\lib
C:\ican50\logicalhost\bootstrap\bin\..\..\bootstrap\config
The service was successfully installed.
C:\ican50\logicalhost\bootstrap\bin>uninstall-nt-service.bat

-----
eGate 5.0 Management Agent Uninstall Bootstrap NT Service
Copyright (c) 2003, SeeBeyond Technology Corporation,
All Rights Reserved
-----

The service was successfully uninstalled.
```

7.4.2 Starting the Logical Host Manually on a Windows System

Before you can manually start a Logical Host you must first create a Project in the Enterprise Designer (see [Enterprise Designer](#) on page 26). After completing a Project, you start the Logical Host by opening a **command prompt** and typing a command on the command line which includes a string of required, optional, and initial (first-time only) command line arguments appended to the startup command. The startup command string uses the following syntax:

```
<path>\logicalhost\bootstrap\bin>bootstrap <arguments>
```

For example, the command

```
c:\eGate50\logicalhost\bootstrap\bin>bootstrap -h
```

displays Help text that explains the command line arguments in detail.

To start the Logical Host

- 1 On your desktop, open a **command prompt**.
- 2 Navigate to **c:\eGate50\logicalhost\bootstrap\bin**.
- 3 To start the Logical Host, on the command line, do one of the following:
 - If you are starting the Logical Host for the first time, run the bootstrap command and include all of the required parameters shown in Table 23.

- If you are starting the Logical Host after having started it at least once previously, do the following:
 - A Navigate to `c:\eGate50\logicalhost\bin`.
 - B Double-click `bootstrap.bat`.

7.4.3 Starting the Logical Host on a UNIX System

To start the Logical Host

- 1 Navigate to the `bootstrap/bin` directory in the location where you installed the Logical Host, for example:

```
cd /home/my_user_name/egate50/logicalhost/bootstrap/bin
```

- 2 Run the bootstrap script using the following command:

```
./bootstrap.sh arguments
```

See [Logical Host Startup Parameters](#) on page 107 for a complete list of the command-line arguments. Additionally, see [Modifying the Logical Host Startup Configuration File](#) on page 108 for instructions on customizing the `logical-host.properties` file.

Note: Remember that the Logical Host service will continue to run until you manually shut it down.

7.4.4 Starting the Logical Host on a Red Hat Linux System

To start the Logical Host

- 1 Navigate to the `bootstrap/bin` directory in the location where you installed the Logical Host, for example:

```
cd /home/my_user_name/egate50/logicalhost/bootstrap/bin
```

- 2 Run the bootstrap script using the following command:

```
./bootstrap.sh arguments
```

- 3 After the bootstrap command is executed, the script prompts you for the RedHat server release number.

See [Logical Host Startup Parameters](#) on page 107 for a complete list of the command-line arguments. Additionally, see [Modifying the Logical Host Startup Configuration File](#) on page 108 for instructions on customizing the `logical-host.properties` file.

7.4.5 Starting the Logical Host on an HP NonStop Server

On HP NonStop Servers, a platform-specific script called LHINIT provides the bootstrap service. This script works with the pathway process to define the primary and backup CPUs for the Logical Host.

To start the Logical Host

- 1 Open a **command prompt** and type the following startup command string:

```
>sh ./admin.sh start <CPU number>
```

For example:

```
>sh ./admin.sh start 2
```

- 2 To check the status of the bootstrap, type the following string:

```
>sh ./admin.sh status <CPU number>
```

Repository Tools

The Repository stores and manages the setup, component, and configuration information for all of your Projects. To take advantage of this design, the Enterprise Designer includes several analysis and archiving tools. This chapter describes how to use each of these tools with your Projects.

Note: Refer to [Repository](#) on page 22 for more details about the Repository's role in eGate Projects.

8.1 Overview

Each of the Repository-related tools covered in this chapter is briefly discussed below:

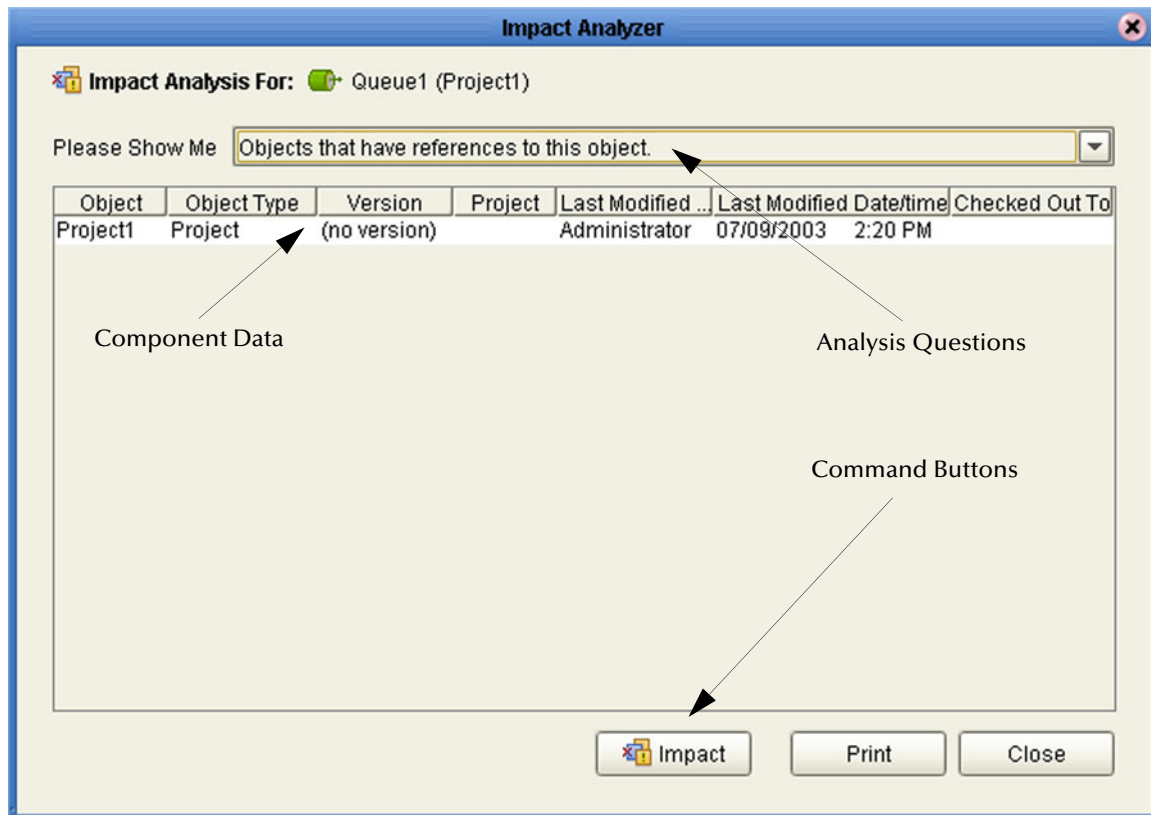
- **Impact Analyzer**
Helps you visualize how a change to one part of a Project would affect the rest of the Project. See [Impact Analysis](#) on page 118.
- **Version Control**
Allows you to maintain multiple versions of Project components. See [Version Control](#) on page 121.
- **Repository Backup/Restore**
Allows you to back up the Repository to an external file and restore a Repository from an external file. See [Repository Backup and Restoration](#) on page 124.
- **Project Export/Import**
Allows you to export a Project from Enterprise Designer to an external file, and import a Project into Enterprise Designer from an external file. See [Project Export and Import](#) on page 126.

8.2 Impact Analysis

The Impact Analyzer tool helps you to determine how changing one part of a Project (or Environment) will affect the rest of the Project. You accomplish this by selecting a Project component and then picking a question to ask about the Project.

Select a Project component, then select **Tools > Impact Analyzer** or click the **Impact Analyzer** toolbar button to display the Impact Analyzer dialog box shown in Figure 90.

Figure 90 Impact Analyzer Dialog Box






The Impact Analyzer is organized into three sections represented by tabs. Each of these tabs is briefly described below:

- **Analysis Questions**
Includes a a drop-down list that contain questions to ask about the displayed component.
- **Component Data**
Lists information about the affected components, including the name of the component, associated Project, who last modified the component, and the date/ time of the last modification.
- **Command Buttons**
Includes four buttons used to execute commands in the dialog box. Refer to [Command Buttons](#) on page 119 for more details.

8.2.1 Command Buttons

Each command button on the Impact Analyzer is briefly described in Table 25.

Table 25 Navigation Buttons

Button	Function
	Select an item from the Component Data list, then click Impact to perform an impact analysis for the that item. See “ Using the Impact Analyzer ” for more information.
	Displays the Print dialog box, which you can use to print the list of component data. Refer to Printing Impact Analysis Results on page 120 for more details.
	Closes the Impact Analyzer dialog box.

8.2.2 Using the Impact Analyzer

This section describes how to use the Impact Analyzer to figure out how changing a Project component impacts other Project components.

To use the Impact Analyzer

- 1 In the Enterprise Explorer, click the **Project Explorer** tab.
- 2 Select a Project component.
- 3 Use one of the following methods to display the **Impact Analyzer** dialog box.
 - ♦ Select **Tools > Impact Analyzer**.
 - ♦ Click the **Impact Analyzer** toolbar button.
- 4 In the *Please show me* drop-down list, select a question to view the Project components that meet the criteria of the question.

Note: Select a Project component on the Component Data list and click **Impact** to see how that component is affected.

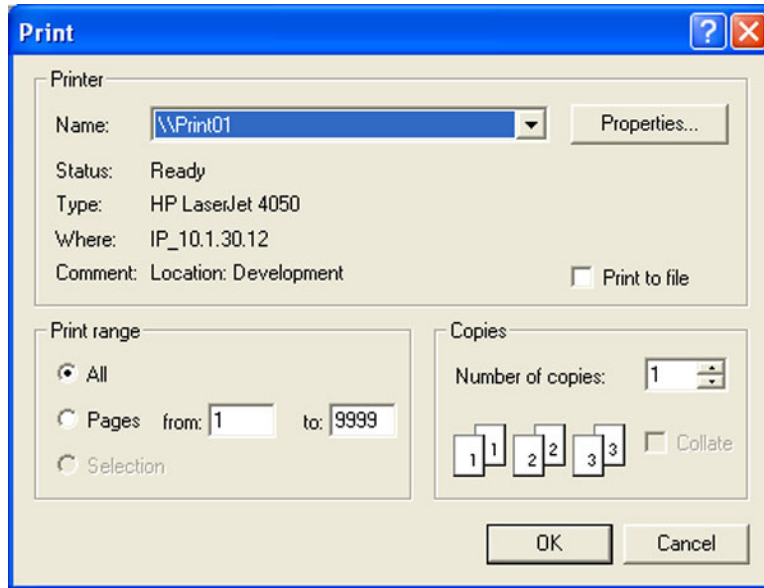
8.2.3 Printing Impact Analysis Results

This section explains how to print the results of component analysis performed with the Impact Analyzer.

To Print Component Data

- 1 From the **Impact Analyzer** dialog box, click **Print** to display the Print dialog box shown in Figure 91.

Figure 91 Print Dialog Box



- 2 Click **OK** to print the component data to your default printer. Your printout will look like Figure 92 shown below.

Figure 92 Printed Component Data

Object	Version	Project	Last Modified By	Last Modified Date/time	Checked Out To
Topic1	(no version)	Test Project	Ben Lee	05/29/2003 11:20 AM	
Queue1	(no version)	Test Project	John Smith	05/29/2003 11:12 AM	
eGate ProjectManager API	(no version)	Unknown	John Smith	05/29/2003 11:12 AM	
Test CM	(no version)	Test Project	Ben Lee	05/29/2003 11:20 AM	
Test Deployment	(no version)	Test Project	John Smith	05/29/2003 11:12 AM	

8.3 Version Control

Version control allows you to maintain several versions and a version history of Project components. Once you have created and configured a Project component, you are able to check that object in. The initial check in occurs when you right-click on a component and select **Check In** from the Component context menu. You use the same method to check out Project components.

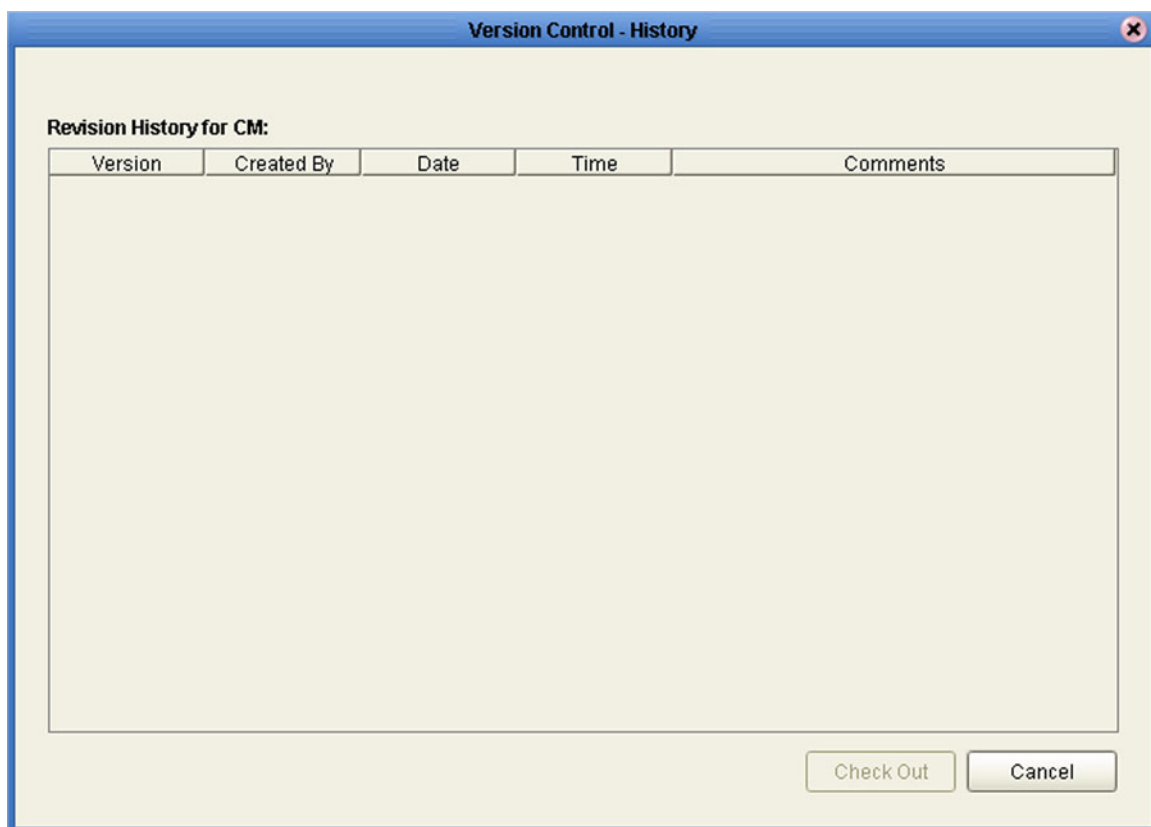
8.3.1 Viewing History of a Project Component

This section describes how to view the version history of a Project component.

To view version history for a Project component

- 1 Click the Project Explorer tab from the Enterprise Explorer.
- 2 Right-click on an icon to display a context menu for that Project component.
- 3 Select **Version History** to display the *Version Control - History* dialog box shown in Figure 93.

Figure 93 Version Control - History Dialog Box



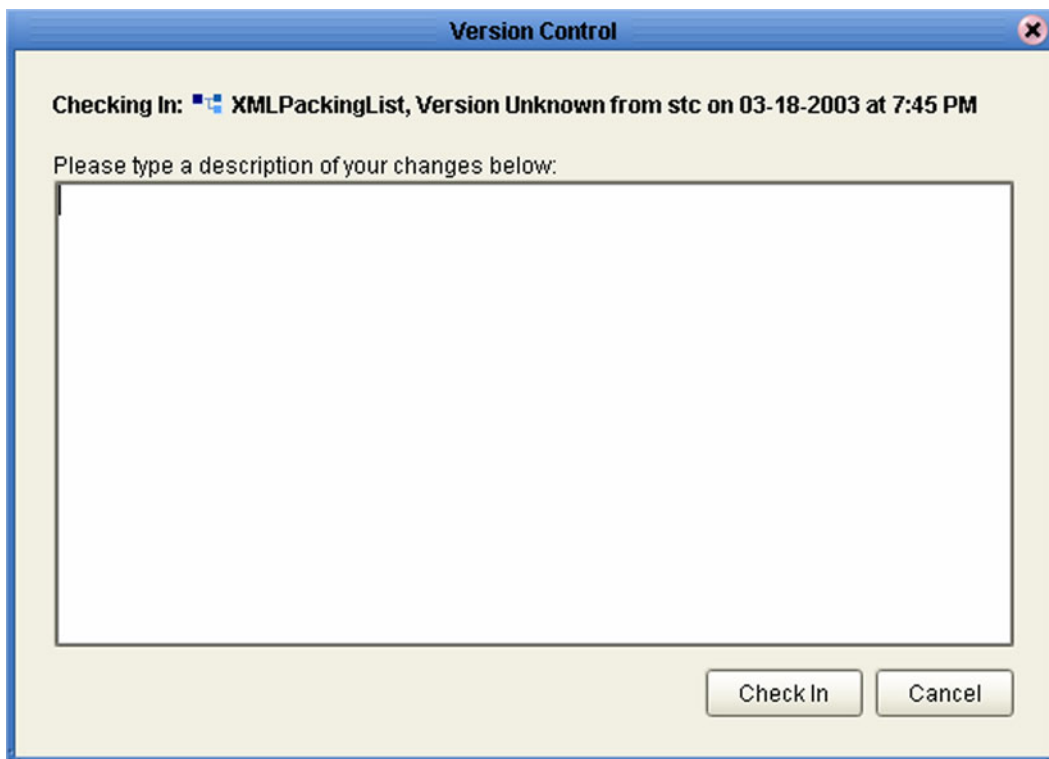
8.3.2 Checking In a Project or Environment Component

This section describes how to check in a version of a Project/Environment component.

To check in a version of a Project/Environment component

- 1 Click the Project/Environment Explorer tab from the Enterprise Explorer.
- 2 Right-click on an icon to display to display the component's context menu.
- 3 Select **Check In** to display the *Version Control* dialog box shown in Figure 94.

Figure 94 Version Control Dialog Box



- 4 Click **Check In** to save your changes to a new version.

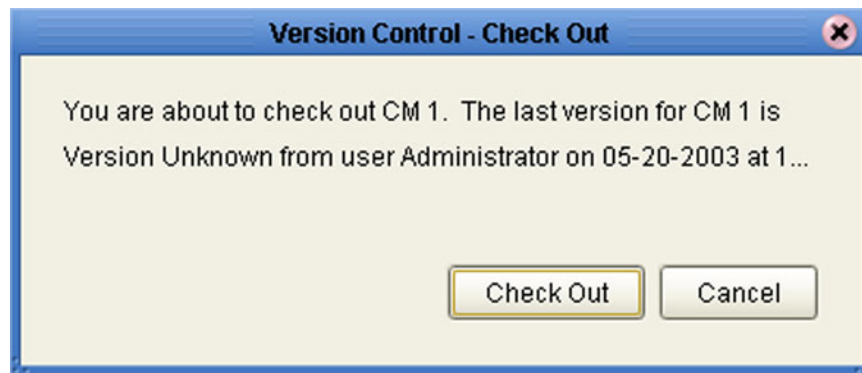
8.3.3 Checking Out a Project or Environment Component

This section describes how to check out a version of a Project/Environment component.

To check out a version of a Project/Environment component

- 1 Click the Project/Environment Explorer tab from the Enterprise Explorer.
- 2 Right-click on an icon to display to display the component's context menu.
- 3 Select **Check Out** to display the *Version Control - Check Out* dialog box shown in Figure 95.

Figure 95 Version Control - Check Out Dialog Box



- 4 Click **Check Out** to open the component.

Note: *Only one user can have a file checked out for editing at a time. If another user attempts to check out the same file, they will receive a message indicating that the file is currently checked out.*

8.4 Repository Backup and Restoration

8.4.1 Backing Up a Repository

The backup function allows you to back up an entire eGate Repository using a command-line script. The backup script creates a backup of all the repository objects and files in the **repository\data** directory including .jar, .nbn, and other binaries, workspaces, users and locks.

During the process of backup the server is locked, so that users are not able to change objects while a backup is in progress. The backup files are .zip files and can be viewed using a decompression utility such as WinZip.

Location of script file:

```
repository/util/backup.bat (or .sh)
```

Command Syntax:

```
backup username password filename
```

Important: *The Repository backup produces a complete snapshot of the Repository, including all installed products—the resulting file, even though compressed, is very large.*

To back up a Repository using the backup script

- 1 Open a command prompt and change directory to *source_repository_root/util*.
- 2 Type (for example): **backup Administrator stc c:/mybackup.zip**.

8.4.2 Restoring a Repository

The restore function allows you to restore an entire eGate Repository using a command-line script. The restore script restores from a backup file. It will wipe out any existing objects/files in the repository and overwrite them with the values from the backup file.

In effect, it will restore the complete snapshot of the repository contained in the backup file including the workspaces, users and locks (checkouts). You can restore the backup to the same repository or a different repository.

Before restoring, the repository server must be running. During the restore process the repository is locked. You **must** restart the repository server after restoring.

Location of script file:

```
repository/util/restore.bat (or .sh)
```

Command Syntax:

```
restore username password filename
```

When restoring a Repository, note that:

- Restoring overwrites the contents of the target Repository.
- After restoring a Repository, you must:
 - A Restart the Repository.
 - B Reactivate all deployments.

To restore a Repository using the restore script

- 1 Open a command prompt and change directory to *target_repository_root/Util*.
- 2 Type (for example): **restore Administrator stc c:/mybackup.zip**.
- 3 Restart the Repository.
- 4 Stop Enterprise Designer.
- 5 Restart Enterprise Designer.

8.5 Project Export and Import

Logs of these processes can be found in *repository_root/logs/repository.log*, where *repository_root* represents the path to your repository root directory.

8.5.1 Exporting a Project

The export function allows you to export an eGate Project to an external file using either the Enterprise Designer or a command-line script.

When exporting a Project, note that:

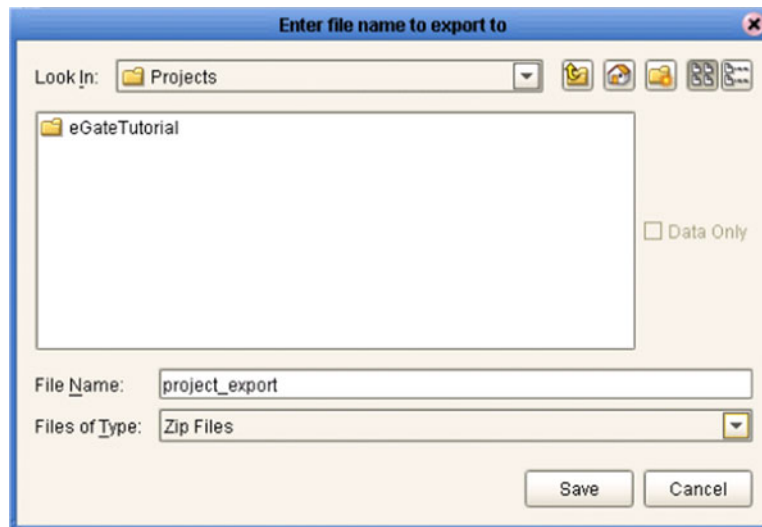
- The exported Project may have references to elements that are in other Projects. A list of such references is generated during the export process.
- Project deployment objects are not exported, because they have references to both Project and Environment elements that are not required at the Project level.

Enterprise Designer Method

To export a Project using Enterprise Designer

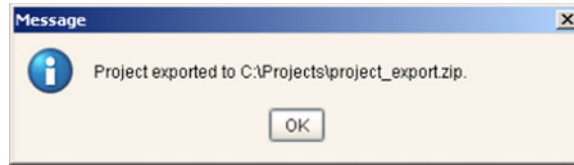
- 1 From the Project context menu, select **Export Project ...** to display the *Enter file name to export to* dialog box shown in Figure 96.

Figure 96 Enter File Name Dialog Box



- 2 Select a location to save the Project file to using the **Look In** drop-down list.
- 3 Enter a name for the Project file in the **File Name** text box.
- 4 Click **Save** to save the Project file. The Message dialog box shown in Figure 97 is displayed when the Project file has been created. (This process takes a few minutes.)

Figure 97 Message Dialog Box



- 5 Click **OK** to close the Message dialog box.

Command-Line Method

You can also export a Project using the following command-line script.

Location of script file:

```
repository_root/util/exportProject.bat (or .sh)
```

Command Syntax:

```
exportProject repositoryURL username password filename projectname
```

To export a Project using the export script

- 1 Open a command prompt and change directory to *export_repository_root/util*.
- 2 Type (for example): **exportProject http://localhost:12345/myrepository Administrator stc c:/myprojectExport.zip myProject1.**

8.5.2 Importing a Project

The import function allows you to import an eGate Project file using the Enterprise Designer.

When importing a Project, note that:

- Existing Projects are not affected by the imported Project.
- During import, if another Project having the same name exists in the target Repository, you will receive an error message and the existing file will not be overwritten.
- You can specify a new Project name and location (in Project Explorer) during import.
- References are validated during import.

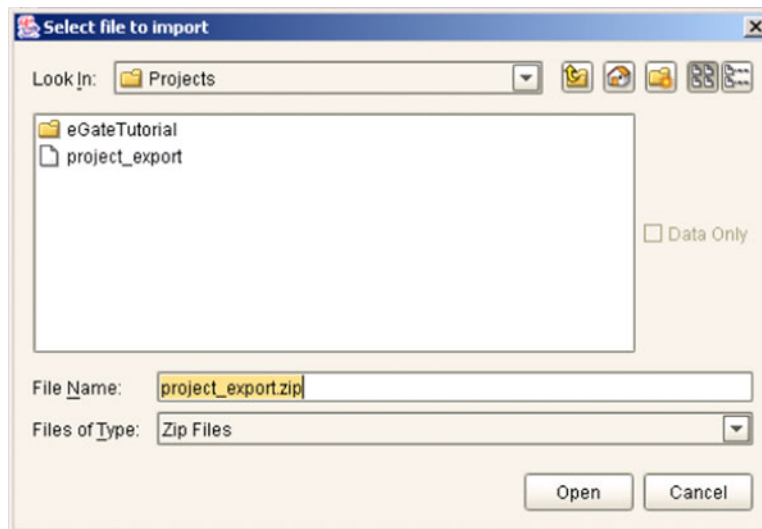
Important: APIs installed in the source Repository must be installed in the Repository into which the Project is imported.

Enterprise Designer Method

To import a Project using Enterprise Designer

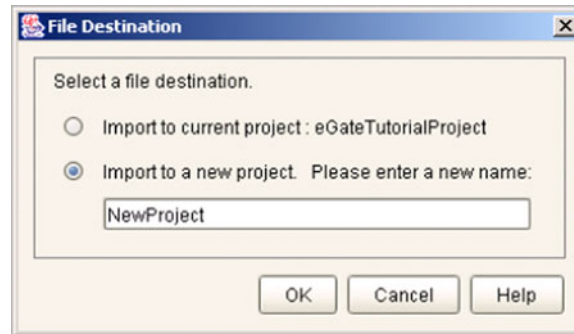
- 1 Create a new Project folder in Project Explorer.
- 2 From the Project context menu, select **Import Project ...** to display the *Select file to import* dialog box shown in Figure 98.

Figure 98 Select File Dialog Box



- 3 Locate and select the Project file that you want to import.
- 4 Click **Open** to import the file.
- 5 A File Destination dialog box will appear as shown in Figure 99, in which you choose whether to import to the same Project name or to a different name.

Figure 99 File Destination Dialog Box



- 6 If you choose to import to a new Project, enter the name for the Project.
- 7 Click **OK** to import the file.
- 8 The Message dialog box shown in Figure 100 displays when the Project file has been imported. (This process takes a few minutes.)

Figure 100 Message Dialog Box



- 9 Click **OK** to close the Message dialog box.

Command-Line Method

You can also import a Project using the following command-line script.

Location of script file:

```
repository_root/util/importProject.bat (or .sh)
```

Command Syntax:

```
"importProject repositoryURL username password filename  
projectname ParentProjectName"
```

Note: If root is the parent, replace **ParentProjectName** with "".

To import a Project using the import script

- 1 Open a command prompt and change directory to `import_repository_root/util`.
- 2 Type (for example): `"importProject http://localhost:12345/myrepository Administrator stc c:/myprojectExport.zip myNewProject1 myExistingProject1"`.

This will take the Project that exists in the export file, rename it to myNewProject1 and attach it as a subproject of myExistingProject1. If the Project is to be attached to the root, then leave the last parameter as an empty string. Make sure you include the quotes ("") around the entire command.

Security

This chapter contains information on the various security features provided in the ICAN Suite.

9.1 Overview

Security features described in this chapter are as follows:

- **Configuration User Management** on page 131 describes the management of users in the ICAN Suite.
- **Environment User Management** on page 134 describes the management of users who would access the applications deployed in an enterprise, using the ICAN Suite.
- **ACL Management** on page 136 describes the management of access control to various components and features in the ICAN Suite.
- **SSL/HTTPS** on page 138 describes the use of Secure Sockets Layer protocols in Web communications.
- **LDAP** on page 142 describes how to configure the ICAN Suite to use an LDAP server for authentication.

9.2 Configuration User Management

This section describes the management of users for the ICAN Suite. These users are able to use the ICAN suite of tools for design, administration, and management of applications in an enterprise.

9.2.1 Roles

The Enterprise Designer allows you to manage user access, based on *roles* and user ID. There are three predefined roles in the ICAN Suite:

- **all**
Users assigned this role can log in to ICAN 5.0. Once logged in, they can inherently connect to the Repository, perform downloads, and access documentation in Enterprise Manager.

Note: The **all** role is assigned to **all users** registered on the system, and provides the most basic, and restrictive, access privileges.

- **administration**
Users assigned this role can log in and connect to the Repository, perform downloads *and uploads*, and access documentation in Enterprise Manager.
- **management**
Users assigned this role can log in and connect to the Repository, perform downloads *but not uploads*, and access documentation in Enterprise Manager. They also can start and stop components using the eGate Monitor.

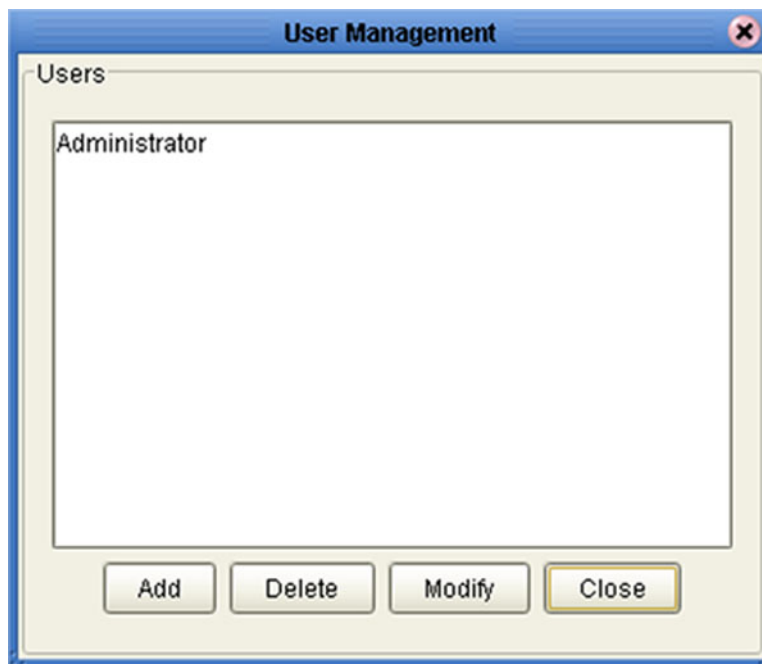
Note: A user can have more than one role, in which case his privileges will be the combined privileges from both roles.

9.2.2 User Management Interface

To open the User Management dialog box

- 1 In the Project Explorer, select the **Repository** icon.
- 2 Right-click to display the Repository context menu.
- 3 Select **User Management** to display the User Management dialog box shown in Figure 101.

Figure 101 User Management Dialog Box

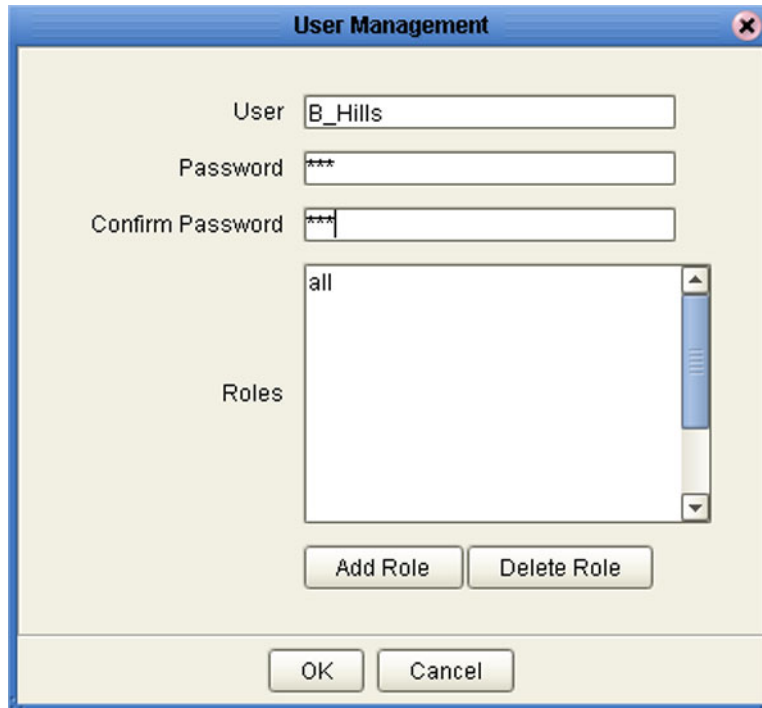


From this dialog box, you can manage user access to the Repository with options for adding, modifying, and deleting users.

To Add a New User

- 1 From the User Management dialog box, click **Add** to display a second User Management dialog box shown in Figure 102.

Figure 102 User Management - Add User



The screenshot shows a dialog box titled "User Management" with a close button (X) in the top right corner. The dialog contains the following elements:

- User:** A text input field containing "B_Hills".
- Password:** A text input field containing three asterisks (***) for masking.
- Confirm Password:** A text input field containing three asterisks (***) for masking.
- Roles:** A list box containing the text "all".
- Buttons:** Below the Roles list box are two buttons: "Add Role" and "Delete Role". At the bottom of the dialog are "OK" and "Cancel" buttons.

- 2 Enter a name for the user in the **User** box. This is the name the user will enter as their login ID during system login.
- 3 Enter a code for the user in the **Password** box. This is the code the user will enter as their password during system login.
- 4 Click the **Add Role** button to add another role for the user.

Note: Every user is automatically assigned to the **all** role, which is required to connect to the Repository; therefore, the **all** role cannot be deleted from the user.

- 5 Click **OK** to create the new user. The newly created user is listed on the User Management dialog box. This user can now access the Enterprise Designer and Repository with the assigned login ID and password.

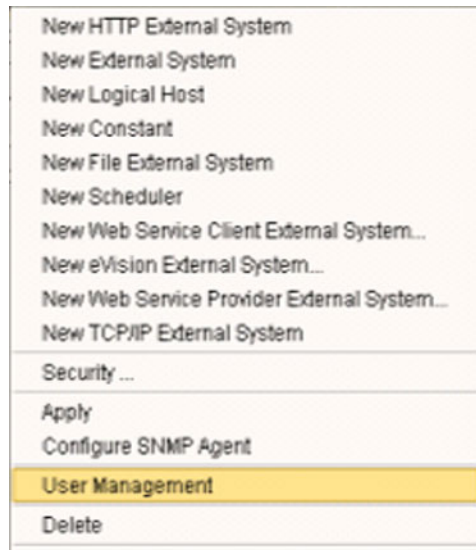
9.3 Environment User Management

This section describes the management of users who would access the applications deployed in an enterprise, using the ICAN suite.

To Open the User Management Dialog Box

- 1 In the Environment Explorer, right-click on an **Environment** to display its context menu (see Figure 103).

Figure 103 Environment Context Menu



- 2 Select **User Management** to display the series of User Management dialog boxes for the Environment, which are identical to those shown in [User Management Interface](#) on page 132. Follow the same procedure described in that section.

9.3.1 Message Server Security

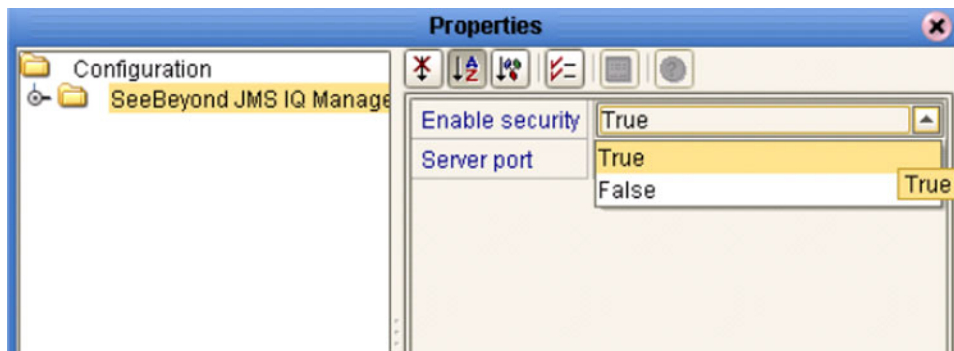
This section describes procedures for enabling security on the message server and clients.

Message Server Settings

To enable message server security

- 1 In the Environment Explorer, right-click on a message server to display its context menu.
- 2 Select **Properties** to display the message server Properties Dialog Box.
- 3 Select **JMS IQ Manager** in the left panel of the Properties Dialog Box to display its properties (see Figure 104).

Figure 104 JMS IQ Manager Properties

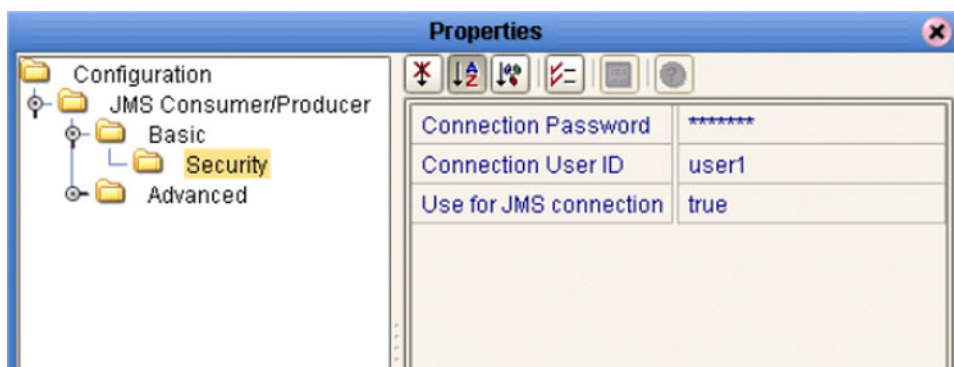


- 4 Select **True** from the drop-down menu to enable security.

JMS Client Security

To enable authentication of Collaborations in JMS Client connections, set the properties as shown in Figure 105.

Figure 105 JMS Client Security Properties



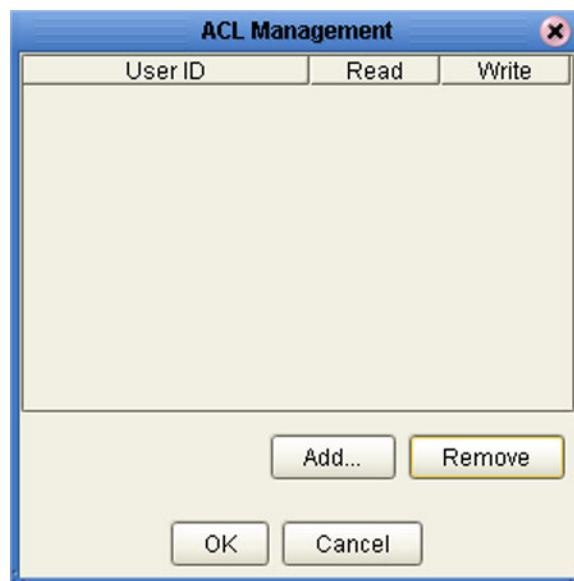
9.4 ACL Management

Access Control List (ACL) Properties allow you to assign read and write access to registered users for a selected object within a Repository. This section describes how to set permissions using the ACL Management dialog box.

To open the ACL Properties dialog box

- 1 From the Project Explorer, select an object icon.
- 2 Right-click to display the object's context menu.
- 3 Select **ACL Management** to display the dialog box shown in Figure 106.

Figure 106 ACL Management Dialog Box

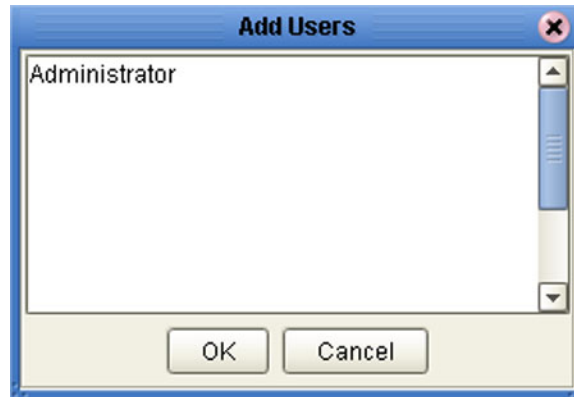


From this dialog box, you can manage user access to the object with options for adding and deleting users and granting them read and write access rights.

To add and assign access rights to a user

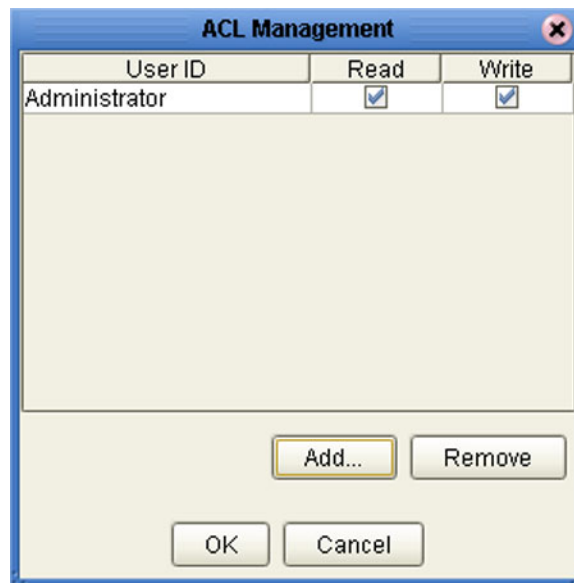
- 1 From the ACL Management dialog box, click **Add** to display the dialog box shown in Figure 107.

Figure 107 ACL Add Users Dialog Box



- 2 Enter the name of the *existing Repository user* to whom you want to grant access to the object into the text box.
- 3 Click **OK** to add the user to the ACL Management list. The user is automatically assigned read access to the object.

Figure 108 ACL Management Dialog Box - Read and Write Access



- 4 Select the **Write** check box for the user if you want them to be able to edit the Project.
- 5 Click **OK** to save your changes to the ACL Properties dialog box.

9.5 SSL/HTTPS

9.5.1 Overview

Secure Socket Layer (SSL) is a technology that allows Web browsers and Web servers to communicate over a secured connection. In this secure connection, the data that is being sent is encrypted before being sent, then decrypted upon receipt and prior to processing. Both the browser and the server encrypt all traffic before sending any data.

Another important aspect of the SSL protocol is *authentication*. During your initial attempt to communicate with a Web server over a secure connection, that server will present your Web browser with a set of credentials in the form of a server certificate. The purpose of the certificate is to verify that the site is who and what it claims to be. In some cases, the server may request a certificate that the client is also who and what it claims to be (which is known as client authentication).

Certificates and Keys

In order to implement SSL, a Web server must have an associated certificate for each external interface, or IP address, that accepts secure connections. The theory behind this design is that a server should provide some kind of reasonable assurance that its owner is who you think it is, particularly before receiving any sensitive information. It may be useful to think of a certificate as a “digital driver's license” for an Internet address. It states with which company the site is associated, along with some basic contact information about the site owner or administrator.

A certificate is a digitally-signed statement from one entity (person, company, etc.), saying that the public key (and some other information) of some other entity has a particular value. When data is digitally signed, the signature can be verified to check the data integrity and authenticity. *Integrity* means that the data has not been modified or tampered with, and *authenticity* means the data indeed comes from whoever claims to have created and signed it.

The certificate is cryptographically signed by its owner and is difficult for anyone else to forge. For sites involved in e-commerce, or any other business transaction in which authentication of identity is important, a certificate can be purchased from a well-known Certificate Authority (CA) such as **Verisign** or **Thawte**.

Certificates are used with the HTTPS protocol to authenticate Web clients. The HTTPS service of the ICAN Repository server will not run unless a server certificate has been installed. Use the procedure outlined below to set up a server certificate that can be used by the ICAN repository server to enable SSL.

The Keytool Utility

One tool that can be used to set up a server certificate is **keytool**, a key and certificate management utility that ships with the J2SE SDK. It enables users to administer their own public/private key pairs and associated certificates for use in self-authentication (where the user authenticates himself/herself to other users/services) or data integrity

and authentication services, using digital signatures. It also allows users to cache the public keys (in the form of certificates) of their communicating peers.

Keytool stores the keys and certificates in a so-called *keystore*. The default keystore implementation implements the keystore as a file. It protects private keys with a password.

The keytool utility enables you to create the certificate. The version that ships with the J2SE SDK programmatically adds a Java Cryptographic Extension provider that has implementations of RSA algorithms. This provider enables you to import RSA-signed certificates.

9.5.2 Installation and Configuration

To install and configure SSL support, you need to perform the following steps:

- 1 Generate a key pair and a self-signed signature.
- 2 Obtain a digitally-signed certificate from a Certificate Authority.
- 3 Import/install the certificate.
- 4 Configure the SERVER.XML file.
- 5 Test the new SSL connection.

The following procedures employ the **keytool** utility.

To generate a key pair and a self-signed signature

- 1 From the command prompt, enter:

```
JAVA_HOME\bin\keytool -genkey -keyalg RSA -alias ICAN -keystore  
keystore_filename
```

Where, for example:

```
keystore_filename = ICAN_HOME\repository\server\conf\ssl\mykeystore
```

- 2 Enter your keystore password: (for example, **seebeyond**)
- 3 The keytool program will ask some questions, such as the following, which you need to answer appropriately:
 - A What is your first and last name?
 - B What is the name of your organizational unit?
 - C What is the name of your organization?
 - D What is the name of your City or Locality?
 - E What is the name of your State or Province?
 - F What is the two-letter country code for this unit?
 - G Is CN=*your_first_and_last_name*, OU=*your_organizational_unit*, O=*your_organization_name*, L=*your_city*, ST=*your_state_or_province*, C=*your_two_letter_country_code* correct?
- 4 Enter key password for ICAN: (RETURN, if same as keystore password)

Note: The example used the following name for the keystore file to be generated: `ICAN_HOME\repository\server\conf\ssl\mykeystore`. You can use this name or another, as long as you use the same name throughout the configuration process.

To obtain a digitally-signed certificate

- 1 From the command prompt, enter the following to generate a Certificate Signing Request (CSR):

```
JAVA_HOME\bin\keytool -certreq -alias ICAN -keyalg RSA -file  
  csr_filename -keystore keystore_filename
```

- 2 Send the CSR for signing.
 - ♦ For example, if you are using Verisign CA, go to <http://digitalid.verisign.com/>. Verisign will send the signed certificate in E-mail.
- 3 Store the signed certificate in a file.

Note: You may skip the following step if you are using only the self-signed certificate. If you are using a self-signed certificate or a certificate signed by a CA that your browser does not recognize, a dialog will be triggered the first time you try to access the server. You can then choose to trust the certificate for this session only, or permanently.

To import the certificate

From the command prompt, enter the following to install the CA certificate:

```
JAVA_HOME\bin\keytool -import -trustcacerts -alias ICAN -file ca-  
  cert_filename -keystore keystore_filename
```

Note: You must have the required permissions to modify the `JAVA_HOME\jre\lib\security\cacerts` file.

An example Connector element for an SSL connector is included in the default `server.xml` file. This Connector element is commented out by default. To enable the SSL Connector for the ICAN repository, remove the comment tags around the SSL Connector element. Changes to the `server.xml` file are read by the ICAN repository server when it is started.

To configure SERVER.XML

- 1 Shut down the ICAN repository server if it is running.
- 2 Open the file `ICAN_HOME/repository/conf/server.xml` using a text editor.
- 3 Find the following section of code in the file (try searching for *Connector*). The following snippet can be put before the first instance of *Connector* in the `server.xml` file.

```
<Connector  
  className="org.apache.catalina.connector.http.HttpConnector"  
  port="8443"  
  minProcessors="5"  
  maxProcessors="75"  
  enableLookups="true"
```

```
    acceptCount="10"  
    debug="0"  
    scheme="https"  
    secure="true"  
  >  
    <Factory  
      className="org.apache.catalina.net.SSLServerSocketFactory"  
      clientAuth="false"  
      protocol="TLS"  
      keystoreFile="ICAN_HOME\repository\server\conf\ssl"  
      keystorePass="seebeyond"  
    />  
  </Connector>
```

- 4 Save and close the file.
- 5 Start the ICAN repository server.

Step 5: Test the new SSL connection

- 1 For testing purposes, and to verify that SSL support has been correctly installed on the ICAN repository server, load the default ICAN repository server introduction page with the following URL:

```
https://localhost:8443/
```

The *https* in this URL indicates that the browser should be using the SSL protocol. The port of 8443 is where the SSL Connector was created in the previous step.

- 2 The first time you load this application, the New Site Certificate dialog displays. Select **Next** to move through the series of New Site Certificate dialogs, and select **Finish** when you reach the last dialog.

Note: You should still have the option to use HTTP to connect to the Enterprise Designer (system administrators should **not** block the HTTP port!).

9.6 LDAP

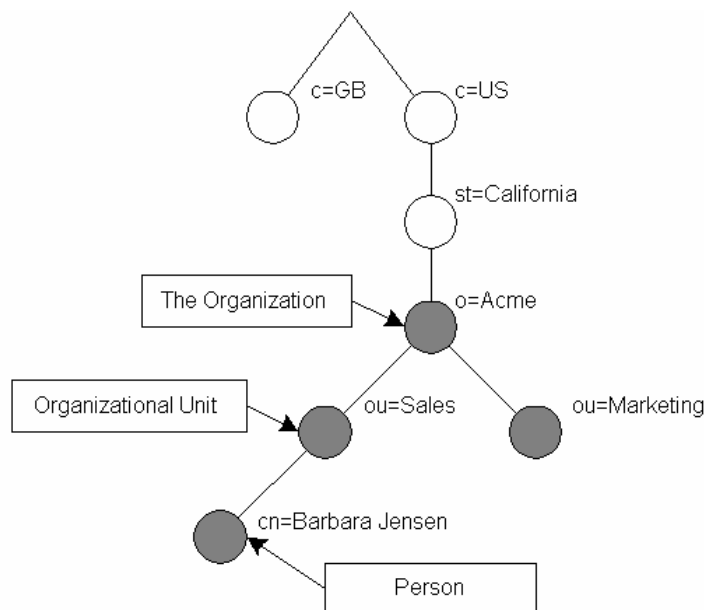
9.6.1 Overview

The Lightweight Directory Access Protocol (LDAP) is a protocol for accessing X.500-based directory services. LDAP runs over TCP/IP or other connection oriented transfer services. LDAP provides a mechanism for a client to authenticate, or prove, its identity to a directory server, paving the way for rich access control to protect the information the server contains. LDAP also supports privacy and integrity security services.

The LDAP information model is based on *entries*. An entry is a collection of attributes that has a globally-unique Distinguished Name (DN). The DN is used to refer to the entry unambiguously. Each of the entry's attributes has a *type* and one or more *values*. The types are typically mnemonic strings, like **cn** for common name, or **mail** for E-mail address. The syntax of values depend on the attribute type. For example, a **cn** attribute might contain the value Babs Jensen. A **mail** attribute might contain the value babs@example.com. A **jpegPhoto** attribute would contain a photograph in the JPEG (binary) format.

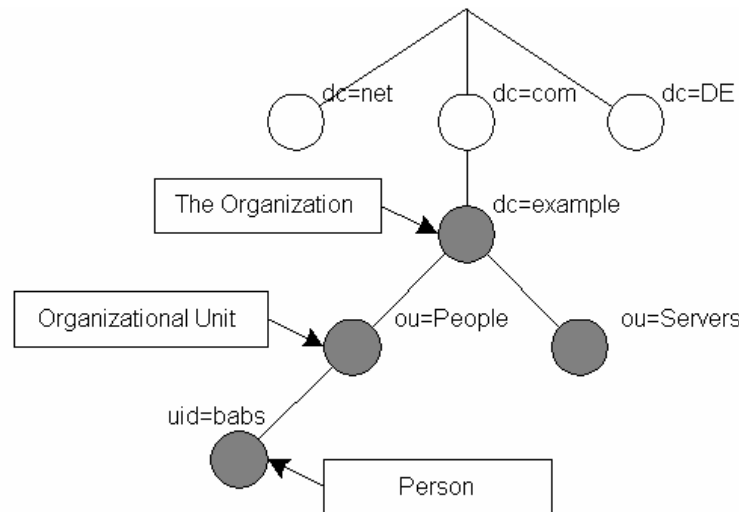
In LDAP, directory entries are arranged in a hierarchical tree-like structure. Traditionally, this structure reflected the geographic and/or organizational boundaries. Entries representing countries appear at the top of the tree. Below them are entries representing states and national organizations. Below them might be entries representing organizational units, people, printers, documents, or just about anything else you can think of. Figure 109 shows an example LDAP directory tree using traditional naming.

Figure 109 LDAP Directory Tree (Traditional Naming)



The tree may also be arranged based upon Internet domain names. This naming approach is becoming increasingly popular, since it allows for directory services to be located using the DNS. Figure 110 shows an example LDAP directory tree using domain-based naming.

Figure 110 LDAP Directory Tree (Internet Naming)



In addition, LDAP allows you to control which attributes are required and allowed in an entry through the use of a special attribute called `objectClass`. The values of the `objectClass` attribute determine the schema rules the entry must obey.

9.6.2 Referencing and Accessing Information

An entry is referenced by its distinguished name (DN), which is constructed by taking the name of the entry itself (called the Relative Distinguished Name or RDN) and concatenating the names of its ancestor entries. For example, the entry for Barbara Jensen in the Internet naming example above has an RDN of `uid=babs` and a DN of `uid=babs,ou=People,dc=example,dc=com`.

LDAP defines operations for interrogating and updating the directory. Operations are provided for adding and deleting an entry from the directory, changing an existing entry, and changing the name of an entry. Most of the time, though, LDAP is used to search for information in the directory. The LDAP search operation allows some portion of the directory to be searched for entries that match some criteria specified by a search filter. Information can be requested from each entry that matches the criteria.

For example, you might want to search the entire directory sub-tree at and below `dc=example, dc=com` for people with the name Barbara Jensen, retrieving the E-mail address of each entry found. On the other hand, you might want to search the entries directly below the `st=California, c=US` entry for organizations with the string Acme in their name, and that have a fax number. LDAP allows you to do either, easily.

9.6.3 Implementing LDAP

The LDAP directory service is based on a client-server model. One or more LDAP servers contain the data making up the directory information tree (DIT). The client connects to servers and asks it a question. The server responds with an answer and/or with a pointer to where the client can get additional information (typically, another LDAP server). No matter which LDAP server a client connects to, it sees the same view of the directory; a name presented to one LDAP server references the same entry it would at another LDAP server. This is an important feature of a global directory service, like LDAP.

9.6.4 Configuring the ICAN Suite to Use LDAP

LDAP servers can be used for both Configuration User Management and Environment User Management.

Configuration User Management

To use an LDAP server with the ICAN system, a *Realm* element needs to be added to the ICAN repository server's `server.xml` file. The Realm element attributes are shown in Table 26.

To add the Realm element to the `server.xml` file

- 1 Open `ICAN_HOME\repository\server\conf\server.xml`.
- 2 Add the following code snippet inside the **ENGINE** tag:

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
        connectionURL="ldap://LDAP_DSN:LDAP_port"
        userPattern="uid={0}, ou=users, o=company.com"
        userRoleName="memberOf"
/>
```

- 3 Save your changes and close the file.
- 4 Start your LDAP server.
- 5 Shut down and restart your ICAN repository server.

Table 26 Realm Element Attributes

Attribute	Parameter	Description/Notes
className		Always use the default className: "org.apache.catalina.realm.JNDIRealm"
connectionURL		Identifies the location of the LDAP server.
	LDAP_port	The port your LDAP server listens for requests; for example, 389.
	LDAP_DSN	The necessary DSN to connect to your LDAP server; for example, 'localhost'.
userPattern		Identifies the Distinguished Name (DN) that represents a user.

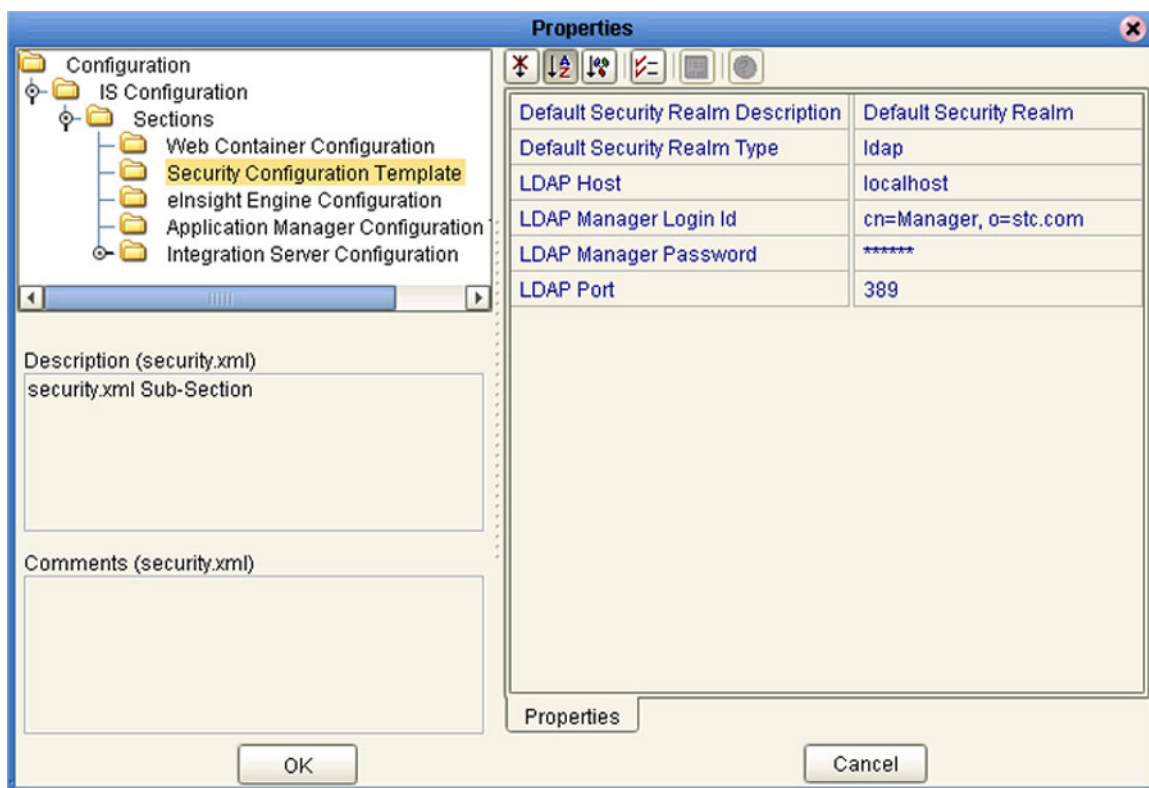
Attribute	Parameter	Description/Notes
userRoleName		The user's globally-unique Distinguished Name (DN) must have an additional attribute that identifies the role(s) to which this user belongs. For example, if the name of the attribute is <i>memberOf</i> , then the ICAN repository server will be configured as userRoleName="memberOf".

Runtime User Management

Integration servers running on the Logical Hosts can be configured to use the LDAP server for their authentication. Run-time authentication for an Integration Server is configured in the Properties Dialog Box for the Security Configuration Template (see Figure 111), where:

- **LDAP Host** is the machine on which the LDAP server is running, for example: *localhost*.
- **LDAP Manager Login Id** is the root distinguished name, for example: *cn=Manager, o=stc.com*.
- **LDAP Manager Password** is the root password, for example: *secret*.
- **LDAP Port** is the port on which the LDAP server is running, for example: 389.

Figure 111 Security Configuration Template Properties Dialog Box



Enterprise Manager

This chapter provides an introduction to the eGate Integrator Enterprise Manager.

10.1 Overview

Enterprise Manager is a specialized integration development environment (IDE) for monitoring and managing eGate Projects and installing product updates. Additional features provided by Enterprise Manager include impact analysis and version control, which are described in [Impact Analysis](#) on page 118 and [Version Control](#) on page 121, respectively.

Important: *Enterprise Manager works only with Microsoft Internet Explorer.*

Note: *Installing Enterprise Manager automatically installs JDK on the client system, whether or not another installation of JDK already exists. The JDK installed with Enterprise Designer is local to the application, and does not interfere with other JDK installations.*

Monitoring and Managing eGate

The Enterprise Manager allows you to monitor and manage eGate components in real-time.

- *Component monitoring* refers to observing the attribute value of a component at specific intervals. Whenever the value crosses a preset threshold, an alert is sent to the Monitoring Server. These alerts are accessible through the Enterprise Manager.
- *Component management* refers to controlling a component's behavior through an external agent—Enterprise Manager.

Product Installation

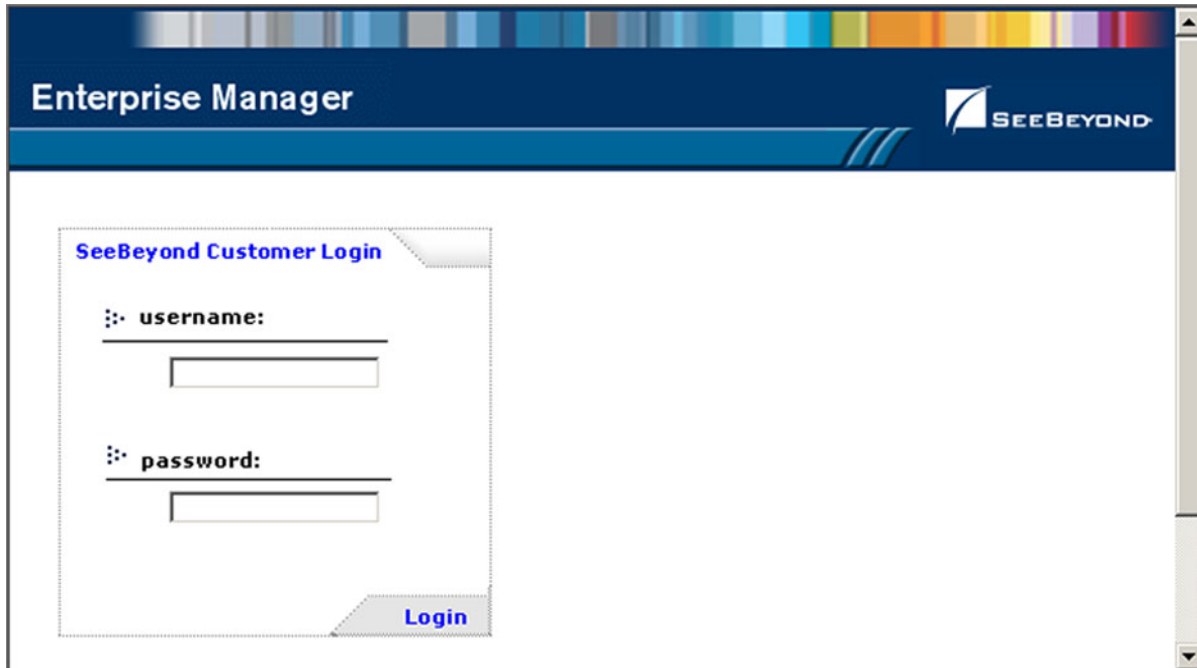
ICAN Suite products are uploaded from the installation media (CD-ROMs) to the Repository Server via the Enterprise Manager. These products are then available to be downloaded and installed from the Repository Server. Uploading and installing products is performed via the *Admin* and *Download* pages in the Enterprise Manager.

10.2 Starting Enterprise Manager

To Start the Enterprise Manager

- 1 Launch Internet Explorer.
- 2 Enter **http://hostname:portnumber** in the **Address** box to display the SeeBeyond Customer Login window shown in Figure 112.

Figure 112 Enterprise Manager Login



Note: The *hostname* is the TCP/IP host name of the server where you installed the Repository. The *portnumber* is the number of the port you entered during installation of the Repository.

- 3 Enter your login ID and password in the **Username** and **Password** boxes and click **Login**.

Figure 113 Enterprise Manager GUI



The Enterprise Manager is organized into three sections represented by tabs (see Figure 113), which are described in the following sections:

- **Home** on page 149
- **Administration** on page 153
- **Downloads** on page 156
- **Documentation** on page 157

To Stop the Enterprise Manager

Click the *Logout* tab (in the upper-right corner of the window) to log out of the Enterprise Manager and return the *SeeBeyond Customer Login window*.

10.3 Home

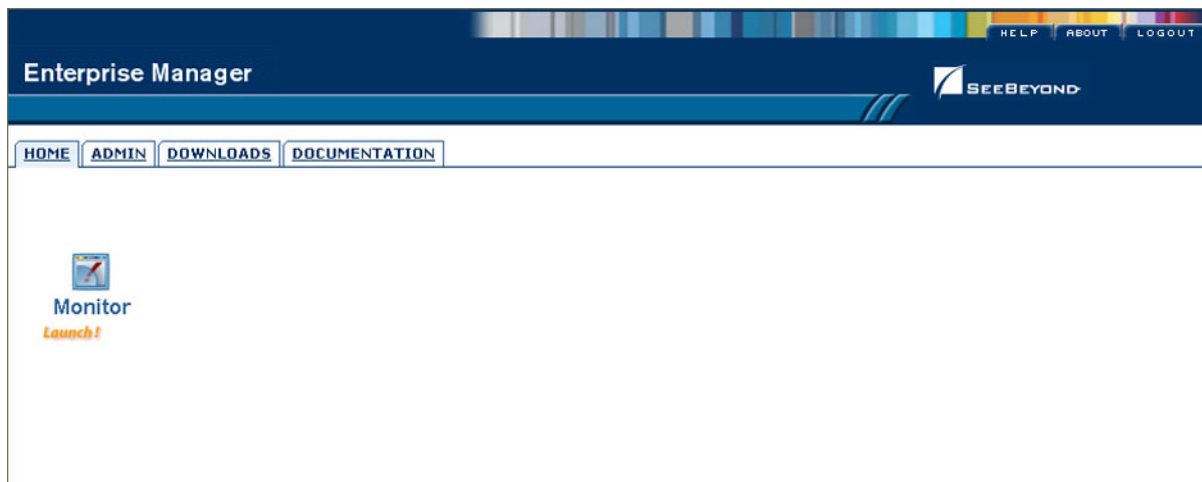
The **Home** tab contains a link to the Monitor Server. This server controls and monitors all the eGate Projects in the Enterprise Designer. Statistics on the status, health, and performance of Project components are available through the Monitor Server.

10.3.1 Monitor Server

To Run the Monitor Server

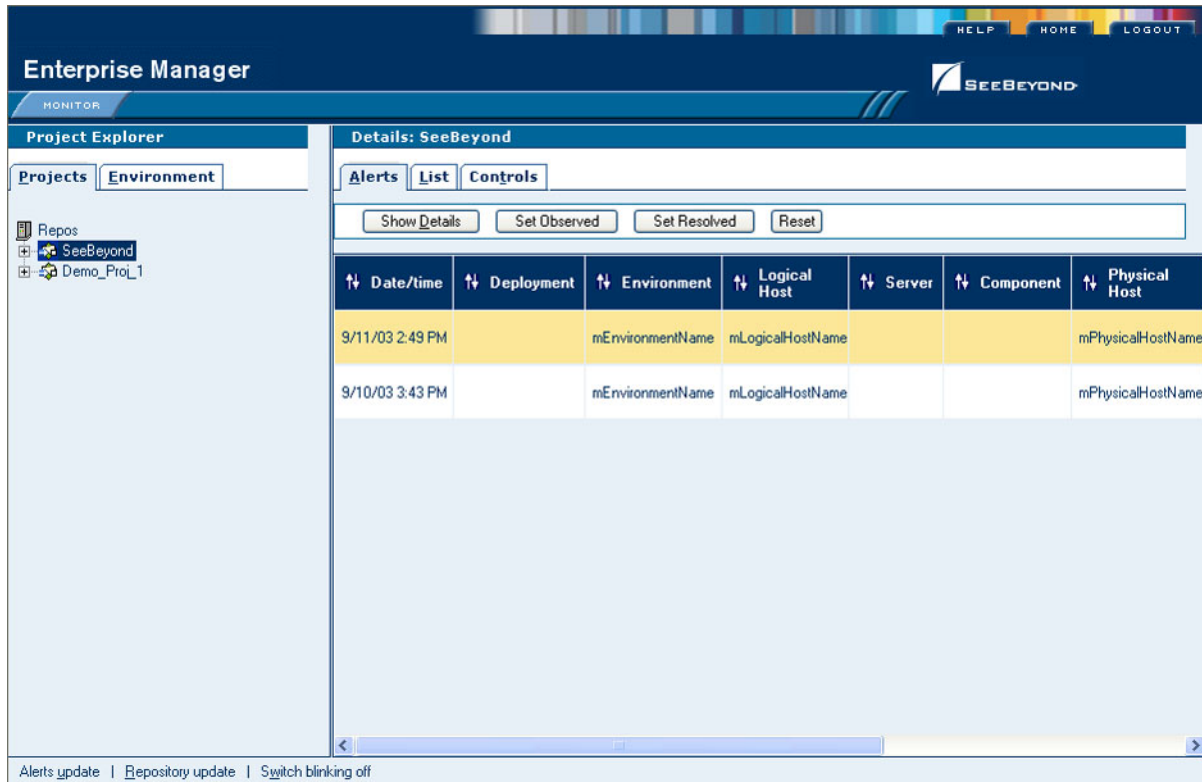
- 1 From the Enterprise Manager GUI, click the **Home** tab to display the Monitor Launch page shown in Figure 114.

Figure 114 Monitor Launch Window



- 2 Click the **Monitor** icon to launch the Monitor page. As with the Enterprise Designer, the Environment Manager contains a Project Explorer and an Environment Explorer.
- 3 Click the **Projects** tab (if not already selected).
- 4 Click any Project icon to display the **Details** area, as shown in Figure 115.

Figure 115 Monitor Page - Projects



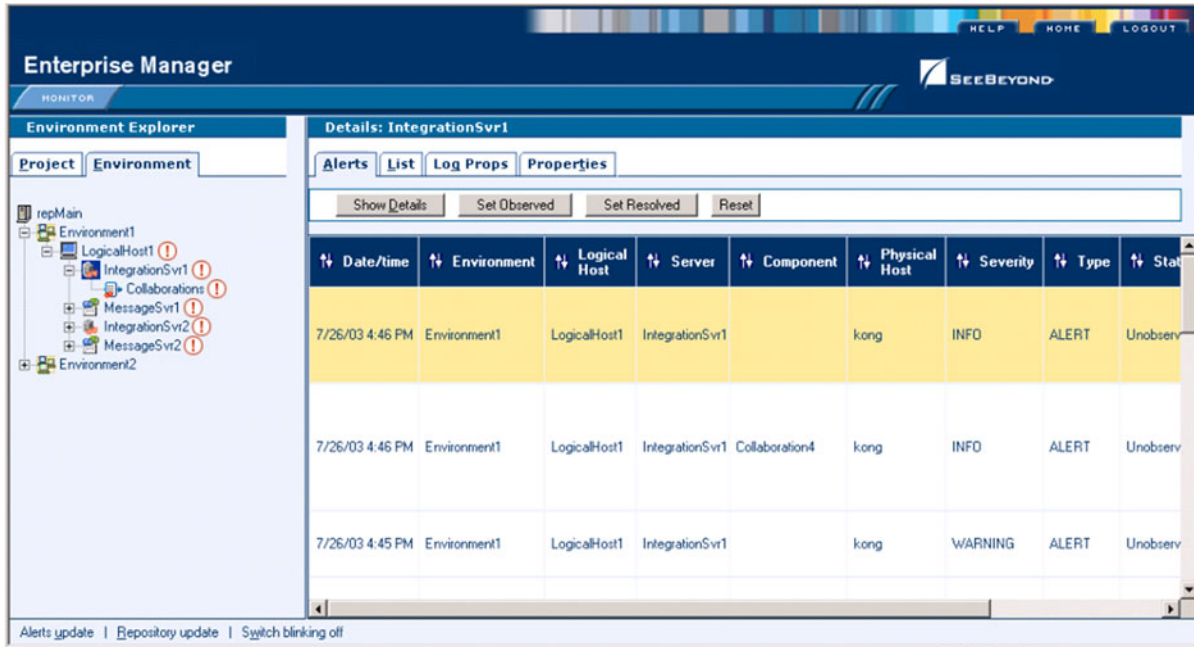
Like the Enterprise Manager, the **Details** area is organized into three sections represented by tabs. Each of these tabs is briefly described below:

- **Alerts**
Displays functionality-related information about the selected Project component.
- **List**
Displays Project component information as a list.
- **Logging**
Displays ...
- **Controls**
Displays ...

Note: Click the **Home** tab (at the top of the window) to exit from the Monitor window and return to the **Admin** tab on the Enterprise Manager.

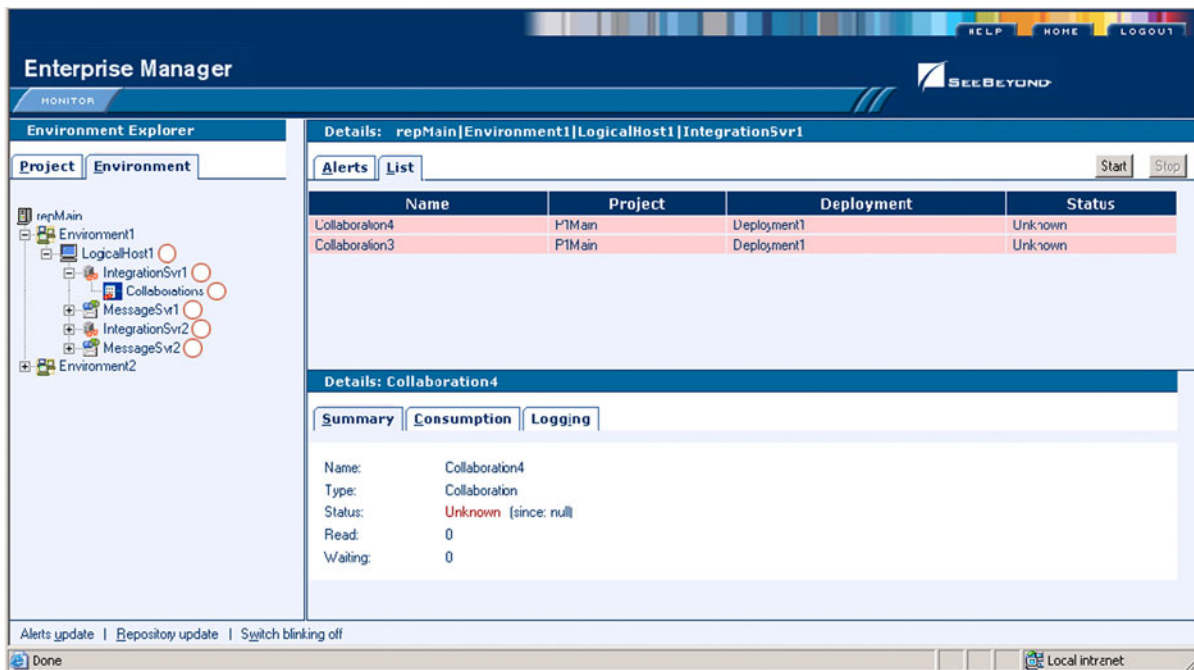
Switching to the Environment Explorer and selecting the Alerts tab in the Details area displays a list of alerts, which can be sorted by different criteria or marked as observed or resolved by means of the command buttons.

Figure 116 Monitor Page - Environment - Alerts



Selecting an Integration Server and clicking *Collaborations* in the Environment Explorer, then selecting the *List* tab in the Details panel, displays all Collaborations on that Host.

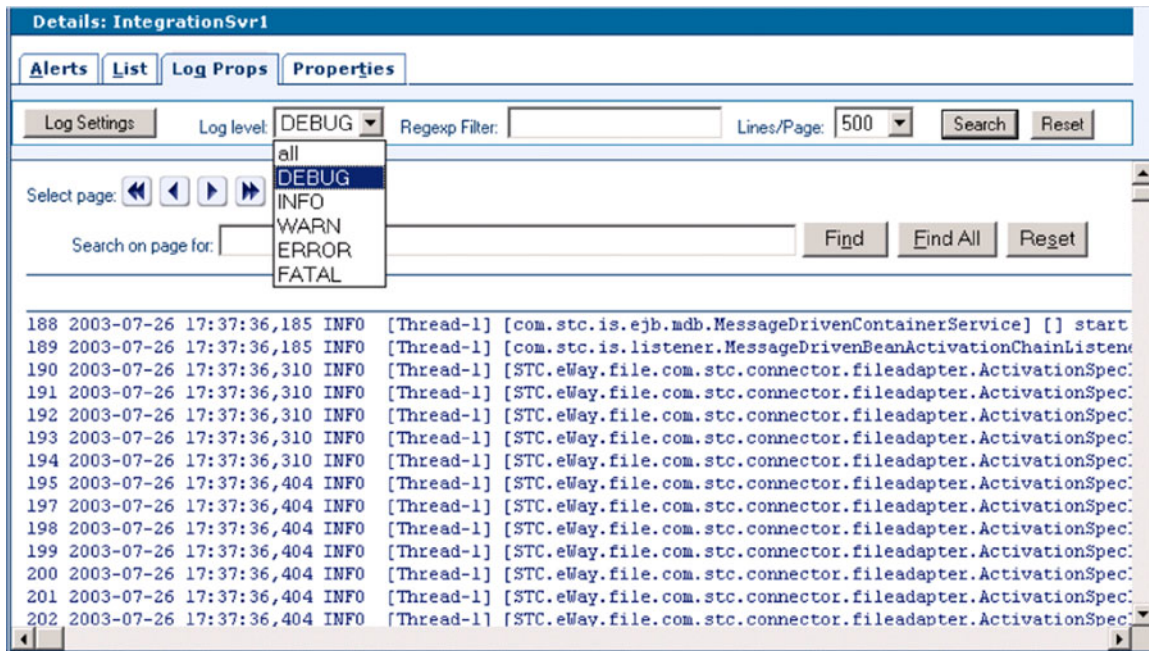
Figure 117 Monitor Page - Environment - List



The Collaborations can be started or stopped from here, and the number of messages waiting or processed can be displayed.

Selecting a Logical Host and clicking the *Log Props* tab in the Details panel displays the log files on the server, which can be sorted and filtered, as well as viewed.

Figure 118 Monitor Page - Environment - Log Props



Additional information on monitoring Projects and viewing logs using the Enterprise Manager Monitor can be found in [Managing Projects](#) on page 158.

10.4 Administration

The **Admin** tab contains tools for accessing the latest eGate 5.0 manifest file and corresponding program files from the SeeBeyond Web site.

To Access the Upload System Component Manifest

- 1 From the Enterprise Manager GUI, click the **Admin** tab. The first time you do this, during installation, the page for selecting the License file is displayed (see Figure 119). **Browse** for the file, then **Submit** for upload.

Figure 119 Upload Product License

Enterprise Manager

HOME ADMIN DOWNLOADS DOCUMENTATION

Select the License file (license.sar) from the CDROM

License File: Browse... Submit

- 2 Once you have uploaded the License file, the Enterprise Manager allows you to select the eGate archive file, as shown in Figure 120.

Figure 120 Upload eGate SAR File

Enterprise Manager

HOME ADMIN DOWNLOADS DOCUMENTATION

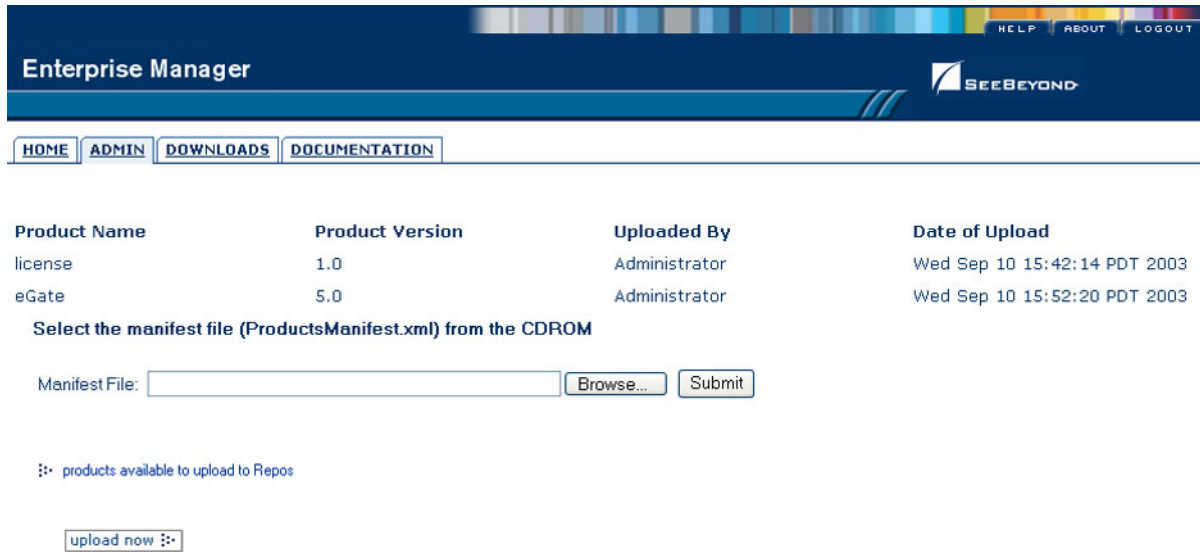
Product Name	Product Version	Uploaded By	Date of Upload
license	1.0	Administrator	Wed Sep 10 15:42:14 PDT 2003

Select the eGate SAR file (eGate.sar) from the CDROM

eGate SAR File: Browse... Submit

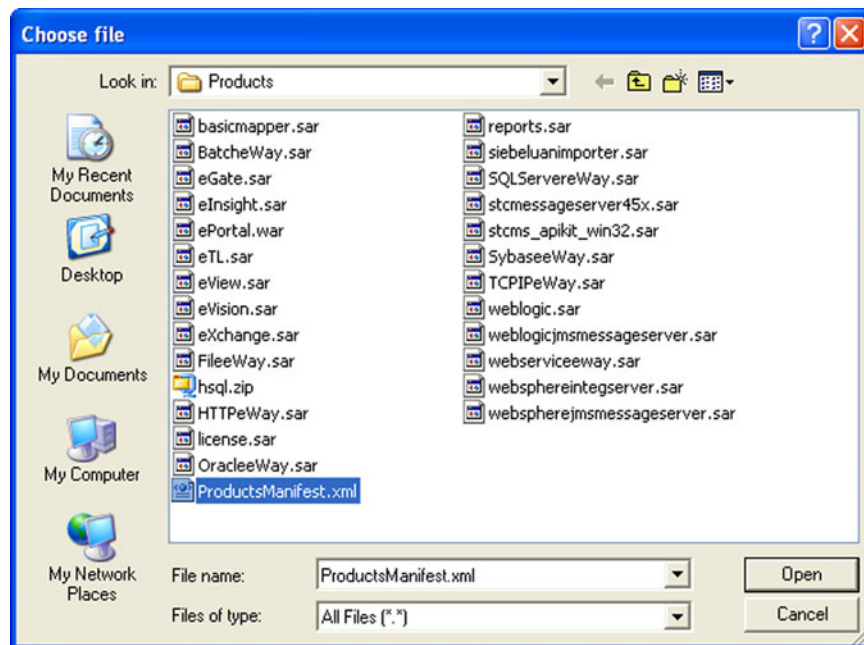
- 3 Once you have uploaded the eGate SAR file, the Enterprise Manager allows you to select the Product Manifest file containing the products included in your license, as shown in Figure 121. If your license and eGate files are already uploaded, this will be the first page you will see.

Figure 121 Upload Product Manifest



- 4 Click **Browse** to display the **Choose File** dialog box shown in Figure 122.

Figure 122 Choose File Dialog Box



- 5 Locate and **Open** the **ProductsManifest.xml** file.

- 6 Click **Submit** to select the manifest file and display the Products Available to Upload list shown in Figure 123.

Figure 123 Products Available to Upload Page

Select the manifest file (ProductsManifest.xml) from the CDROM

Manifest File:

products available to upload to Repos

SeeBeyond Product Suite

Products	Version		
eGate	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eVision	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eInsight	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eView	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eIndex	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eTL	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
eXchange	5.0	<input type="text"/>	<input type="button" value="Browse..."/>
license	5.0	<input type="text"/>	<input type="button" value="Browse..."/>

- 7 Select the products you want to install and click **Upload Now**.

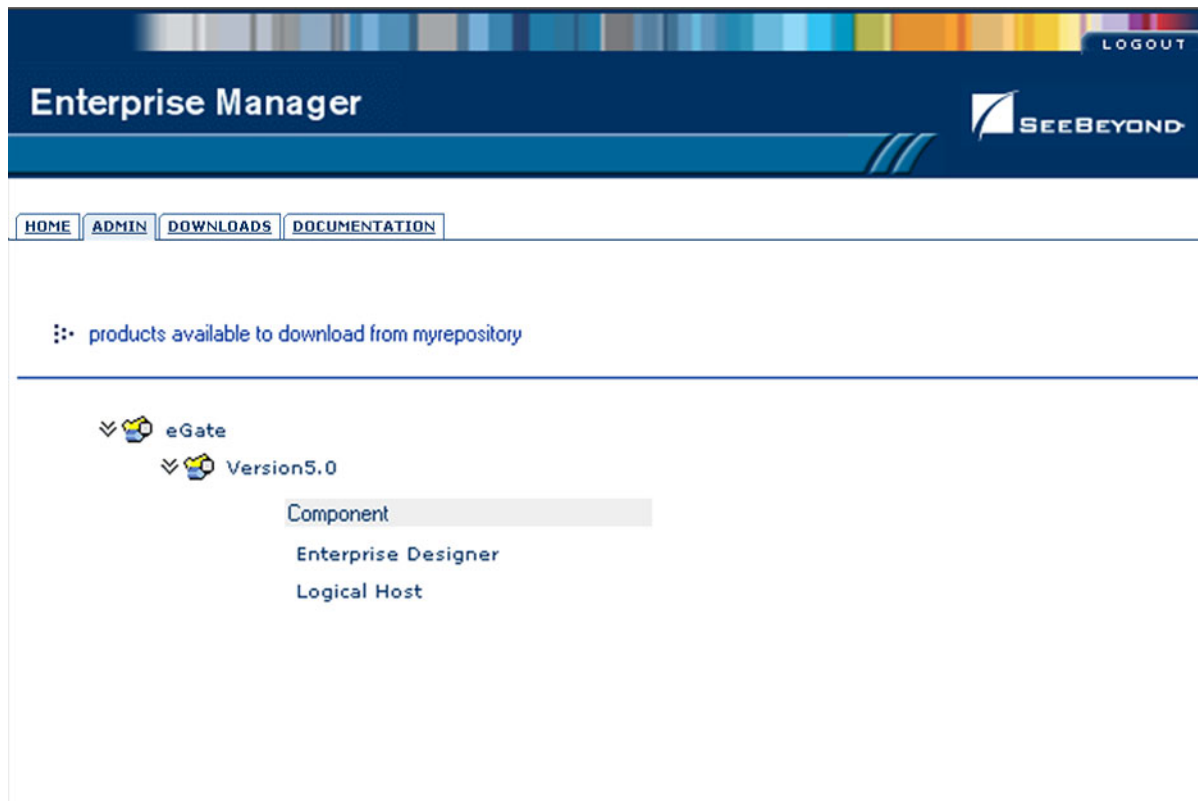
10.5 Downloads

The **Downloads** tab contains links to the latest versions of the Enterprise Designer and Logical Host.

To View the Products Available to Download

- 1 From the Enterprise Manager GUI, click the **Downloads** tab to display the Products Available to Download window shown in Figure 124.

Figure 124 Products Available to Download Page



The Products Available to Download window includes the following products:

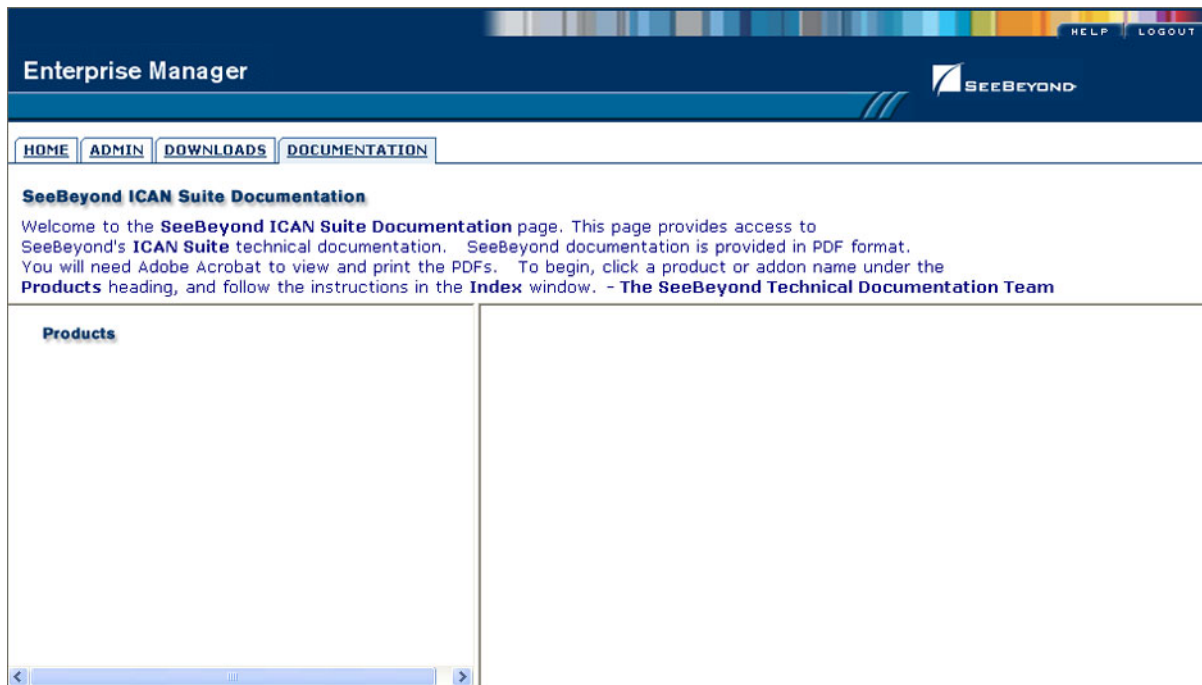
- **Enterprise Designer**
Displays the **File Download** dialog box for the **edesigner.zip** file, which contains the Enterprise Designer program files.
- **Logical Host**
Displays the **File Download** dialog box for the **logicalhost.zip** file, which contains the Logical Host program files.

For comprehensive information on product installation, see the *eGate Integrator Installation Guide*.

10.6 Documentation

The **Documentation** tab contains links to the latest versions of the SeeBeyond ICAN documentation in PDF format. You need to have Adobe Acrobat Reader installed to view or print these documents. Figure 125 shows the initial Documentation page.

Figure 125 Documentation Tab



Note: You must download the documentation SAR files from the installation disk before you can access any documents from this page.

Managing Projects

This chapter describes how to use the Enterprise Manager's facilities for monitoring Projects, and viewing and editing logs and Alerts.

11.1 Overview

Enterprise Manager contains extensive facilities for monitoring and managing your Projects, including the viewing and editing of logs. This chapter includes the following topics:

- **Accessing the Monitor Server** on page 159 shows the initial Enterprise Monitor page, with the Launch Monitor icon.
- **Monitoring Collaborations** on page 160 describes the various ways you can monitor the Collaborations and control various components in your Project.
- **Viewing Alerts** on page 166 describes how to view and set the status of Alerts.
- **Viewing Logs** on page 168 briefly describes how to view, sort, search, and filter messages in the log files.
- **Setting Log Levels** on page 170 describes how to set logging levels in Enterprise Manager.

11.2 Accessing the Monitor Server

The Enterprise Manager's **Home** tab contains a link to the Monitor Server. Click the **Monitor** icon to launch the Monitor page.

Figure 126 Monitor Launch Window



The Monitor Server interface offers the following viewing controls:

- ALT and drag the cursor to scroll.
- CTRL and click to zoom out.
- CTRL-SHIFT and click to zoom in.

11.3 Monitoring Collaborations

Enterprise Manager has extensive monitoring capabilities for Projects, one of the most useful being for Collaborations. To demonstrate these capabilities, we shall take an example Project whose Connectivity Map is shown in Figure 127.

Figure 127 Example Project Connectivity Map



When you launch the Monitor Server, the Environment Explorer is displayed by default. If you select an Integration Server and click **Collaborations**, the Details: List tab will display all Collaborations on that Logical Host (see Figure 128).

Figure 128 Monitor - Details - List Collaborations

The screenshot shows the Enterprise Manager interface. On the left is the Environment Explorer with a tree view showing Repository, Environment1, LogicalHost1, IntegrationSvr1, Collaborations, MessageSvr1, Topics, and Queues. The main pane displays the 'Details: Repository|Environment1|LogicalHost1|IntegrationSvr1' view with a 'List' tab selected. Below the tabs is a table with the following data:

Name	Project	Deployment	Status
Collaboration1	Test1	Test1DP	Running
Collaboration2	Test1	Test1DP	Running
Collaboration3	Test1	Test1DP	Running

Note: When a Project is first started, or freshly activated after de-activation, the status will appear as **Unknown** until the first message is sent. The status will then change to **Running**.

By selecting a Collaboration in the list, you can start or stop the Collaboration by means of the buttons above the tab (see Figure 129) or by right-clicking on the component in Environment Explorer and using the resulting menu (see Figure 130). By default, a Collaboration summary is displayed in the Collaboration detail panel below the list.

Figure 129 Starting and Stopping Collaborations - Details Panel

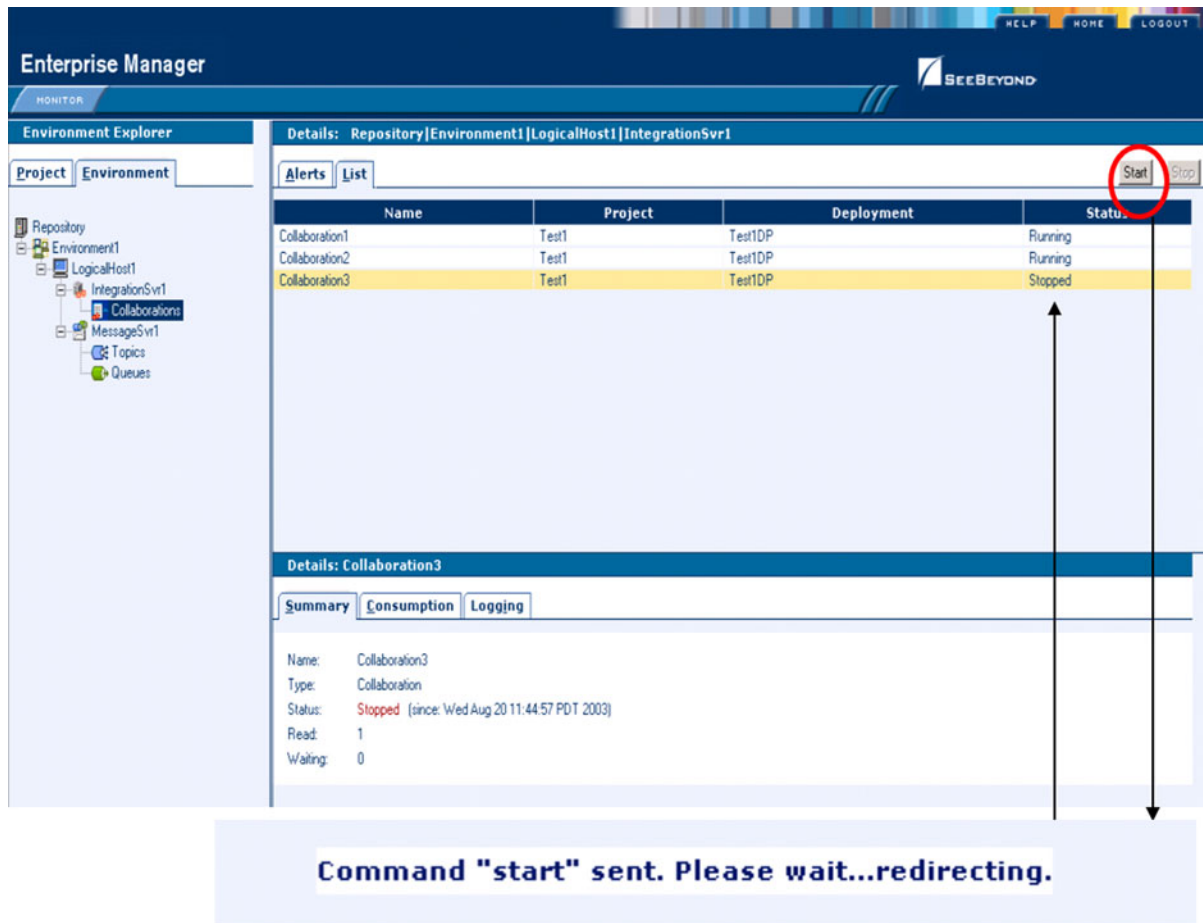
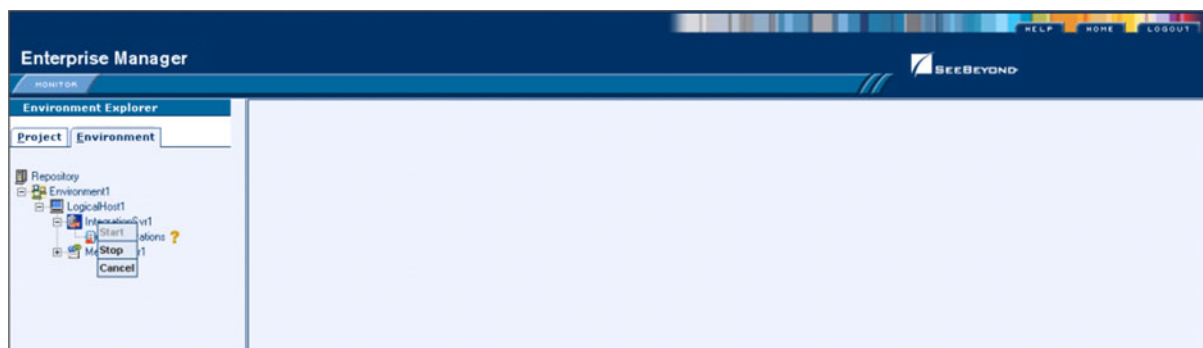
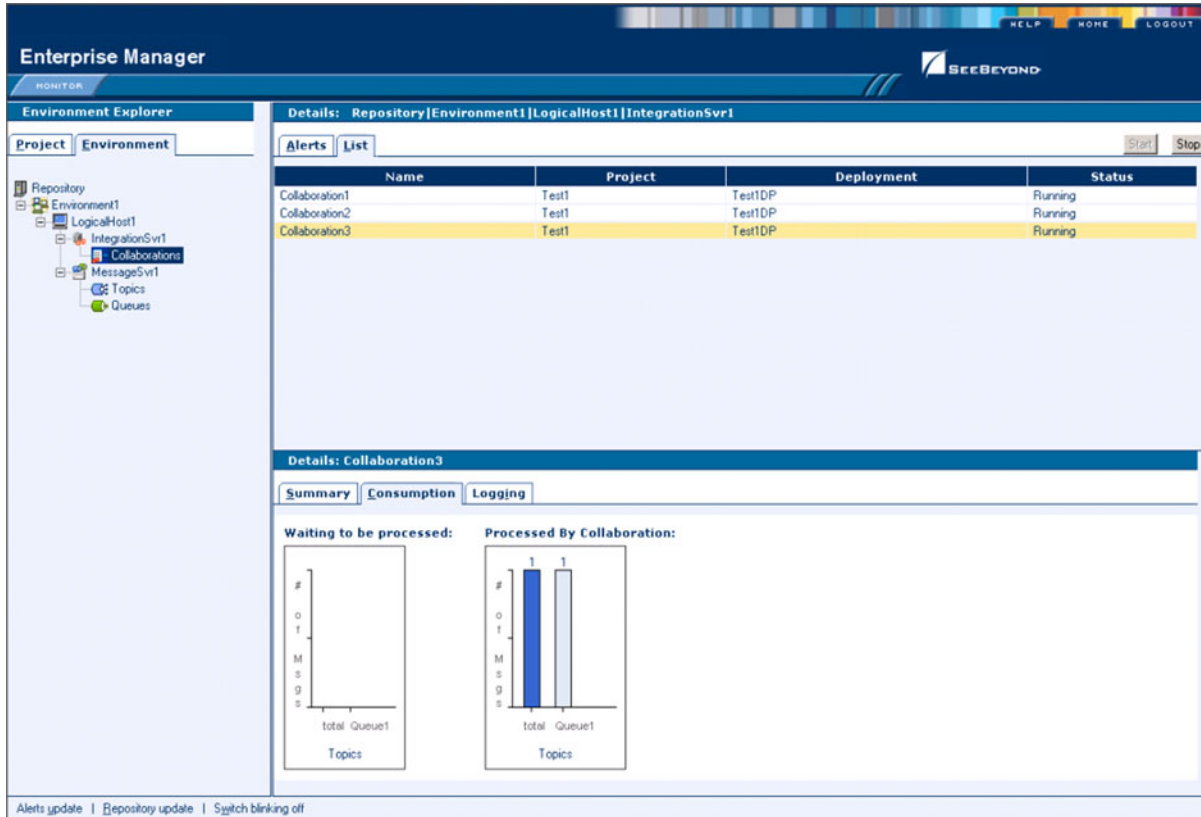


Figure 130 Starting and Stopping Collaborations - Explorer



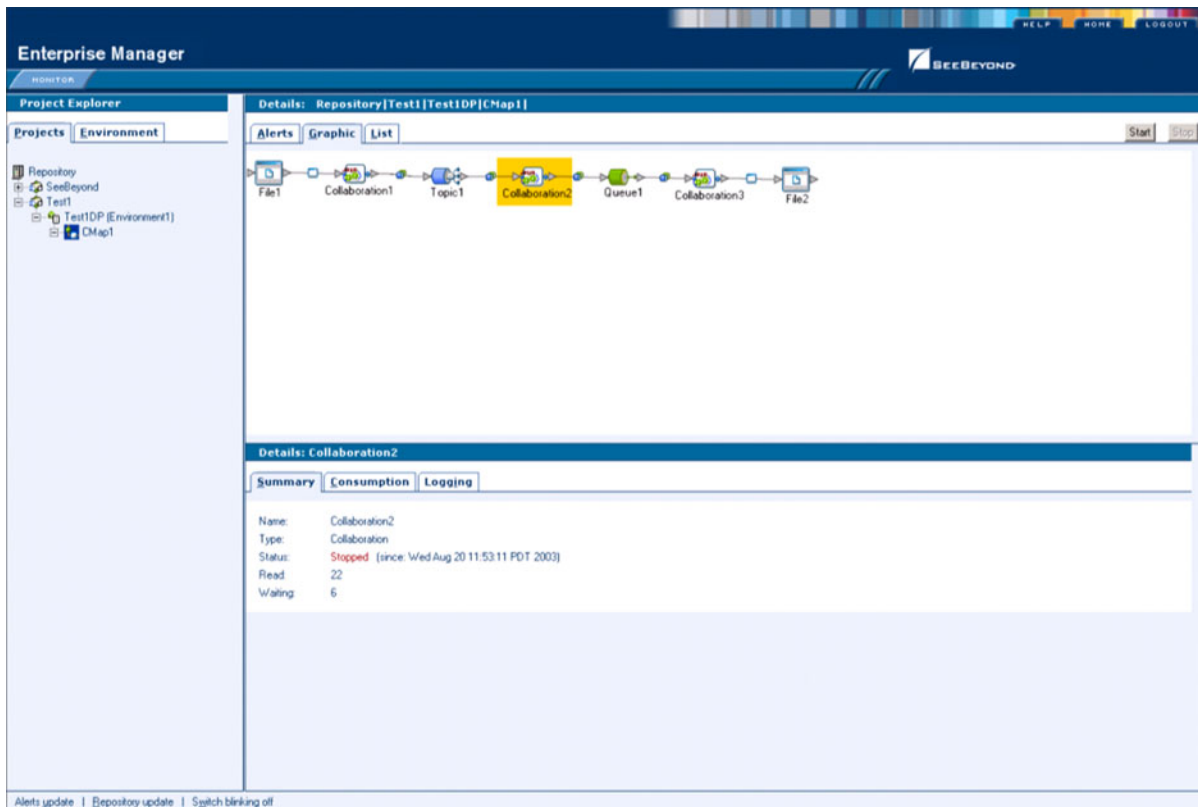
If you select the **Consumption** tab, you will see the number of pending and processed messages in graphical form (see Figure 131).

Figure 131 Pending and Processed Messages



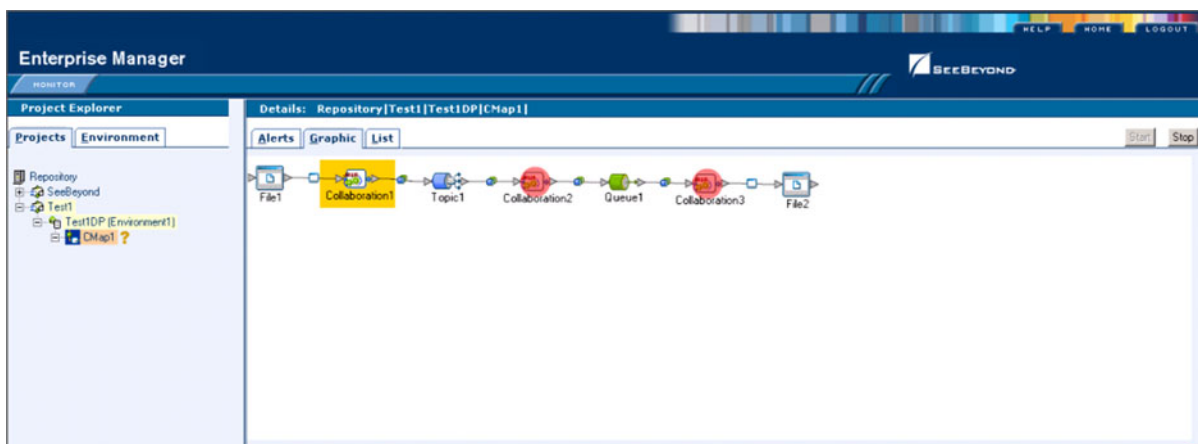
An alternative way to monitor Collaborations is to use the Projects Explorer, which displays all existing Deployment Profiles and Connectivity Maps. You will see the same Connectivity Map as in the Enterprise Designer, with the currently selected Collaboration highlighted in yellow (see Figure 132).

Figure 132 Monitor - Project Explorer



Additionally, non-active Collaborations are highlighted in red, making them immediately visible (see Figure 133).

Figure 133 Active/Non-active Collaborations



As with the monitoring option using the Environment Explorer (shown in [Figure 131 on page 162](#)), selecting a specific Collaboration in the Connectivity Map will display details in the lower panel (see [Figure 134](#) and [Figure 135](#)).

Figure 134 Pending and Processed Messages

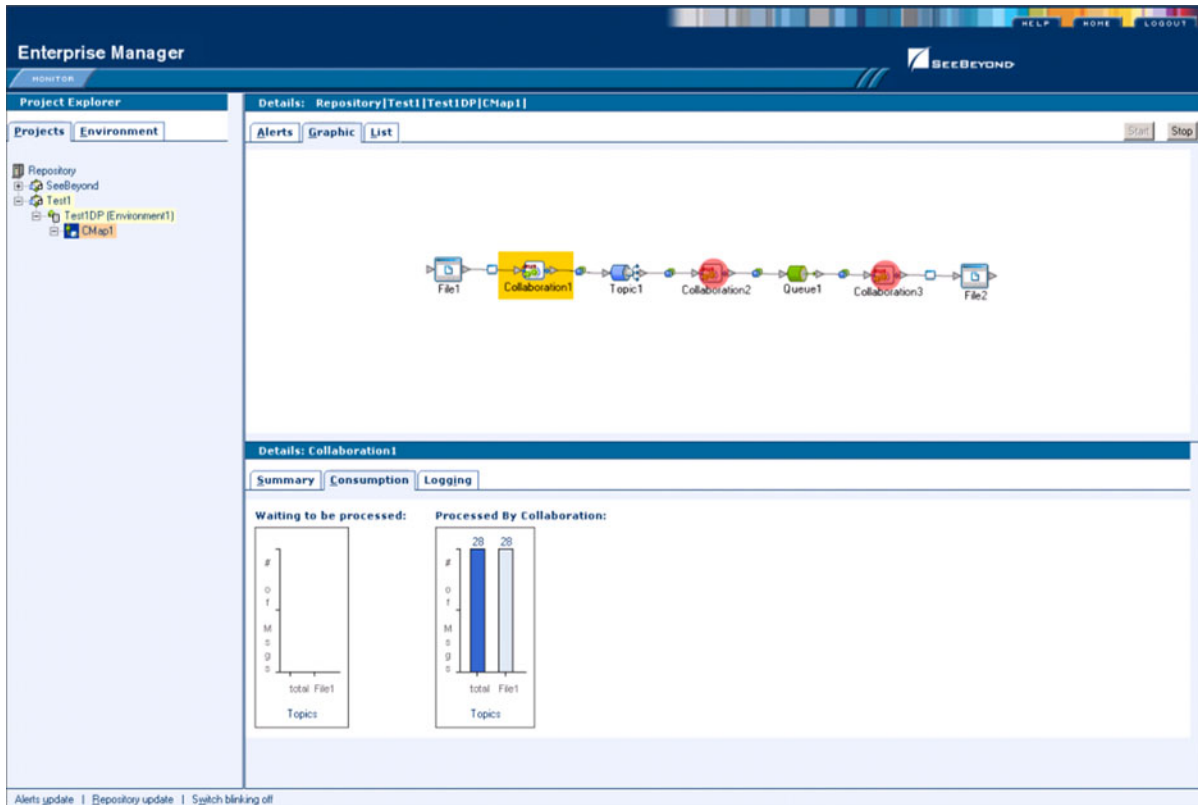
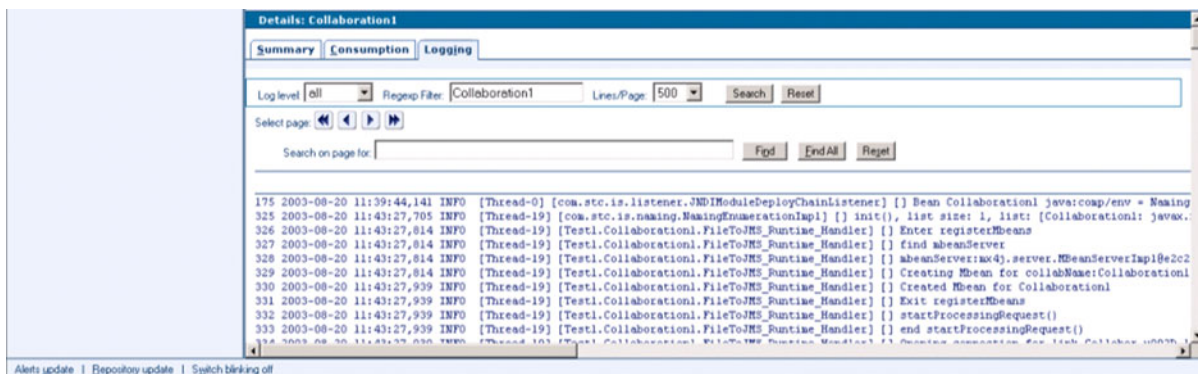


Figure 135 Logging Details



If you select a Connectivity Map in Project Explorer, you can see the status of all components of that Connectivity Map (see Figure 136).

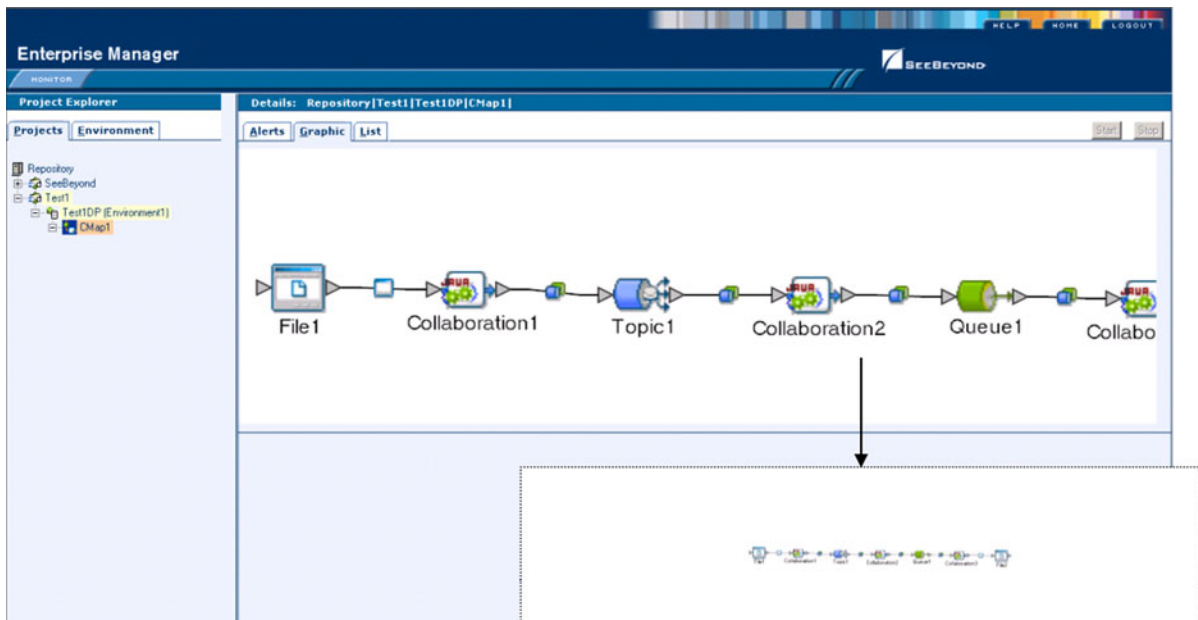
Figure 136 Connectivity Map Details: Components



You can scroll and zoom in or out on a Connectivity Map display using the following commands (see Figure 137):

- ALT and drag the cursor to scroll.
- CTRL and click on the Connectivity Map to zoom out.
- CTRL-SHIFT and click on the Connectivity Map to zoom in.

Figure 137 Connectivity Map View - Zoom In/Out



11.4 Viewing Alerts

Selecting a Collaboration and clicking the **Alerts** tab in the upper Details panel displays a list of all alerts for the selected Collaboration (see Figure 138). These can be sorted by different criteria, or marked as observed or resolved.

Figure 138 Collaboration Alerts

Date/Time	Environment	Logical Host	Server	Component	Physical Host	Severity	Type	Status	State	Description
8/20/03 11:45 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.
8/20/03 11:44 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Unobserved	Stopped	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is stopped.
8/20/03 11:43 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.
8/20/03 11:43 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration2	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration2 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.

By clicking the **Set Resolved** button, you change the status of the selected Alert to *Resolved* (see Figure 139).

Figure 139 Alert Status

Date/Time	Environment	Logical Host	Server	Component	Physical Host	Severity	Type	Status	State	Description
8/20/03 11:45 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.
8/20/03 11:44 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Resolved	Stopped	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is stopped.
8/20/03 11:43 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration3	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.
8/20/03 11:43 AM	Environment1	LogicalHost1	IntegrationSrv1	Collaboration2	aloh_dell	INFO	ALERT	Unobserved	Running	Collaboration Collaboration2 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_dell is running.

By clicking the **Show Details** button, you can see all details for the selected Alert (see Figure 140).

Figure 140 Alert Details

The screenshot displays the Enterprise Manager interface. On the left is the Environment Explorer tree showing a hierarchy: Repository > Environment1 > LogicalHost1 > IntegrationSrv1 > Collaborations. The main area shows a table of alerts. The 'Show Details' button for the selected alert is circled in red. An 'Alert Details' dialog box is open, showing the following information:

Alert Details	
Date/Time	8/20/03 11:44 AM
Environment Name	Environment1
LogicalHost Name	LogicalHost1
Server Name	IntegrationSrv1
Component Project Name	Test1
Component Name	Collaboration3
Physical Host Name	aloh_del
Severity	INFO
Type	ALERT
Status	Unobserved
State	Stopped
Description	Collaboration Collaboration3 under Project Test1 on IntegrationSrv1 on LogicalHost1 in Environment1 on aloh_del is stopped.
Details	Collaboration Collaboration3 is STOPPED

11.5 Viewing Logs

11.5.1 Integration Server Level

In the Environment view, select an Integration Server (or Logical Host) and click the **Logging** tab in the Details panel to display all log messages for the selected component (see Figure 141). You can also filter the list for a specific log level (see Figure 142).

Figure 141 Integration Server Log Messages

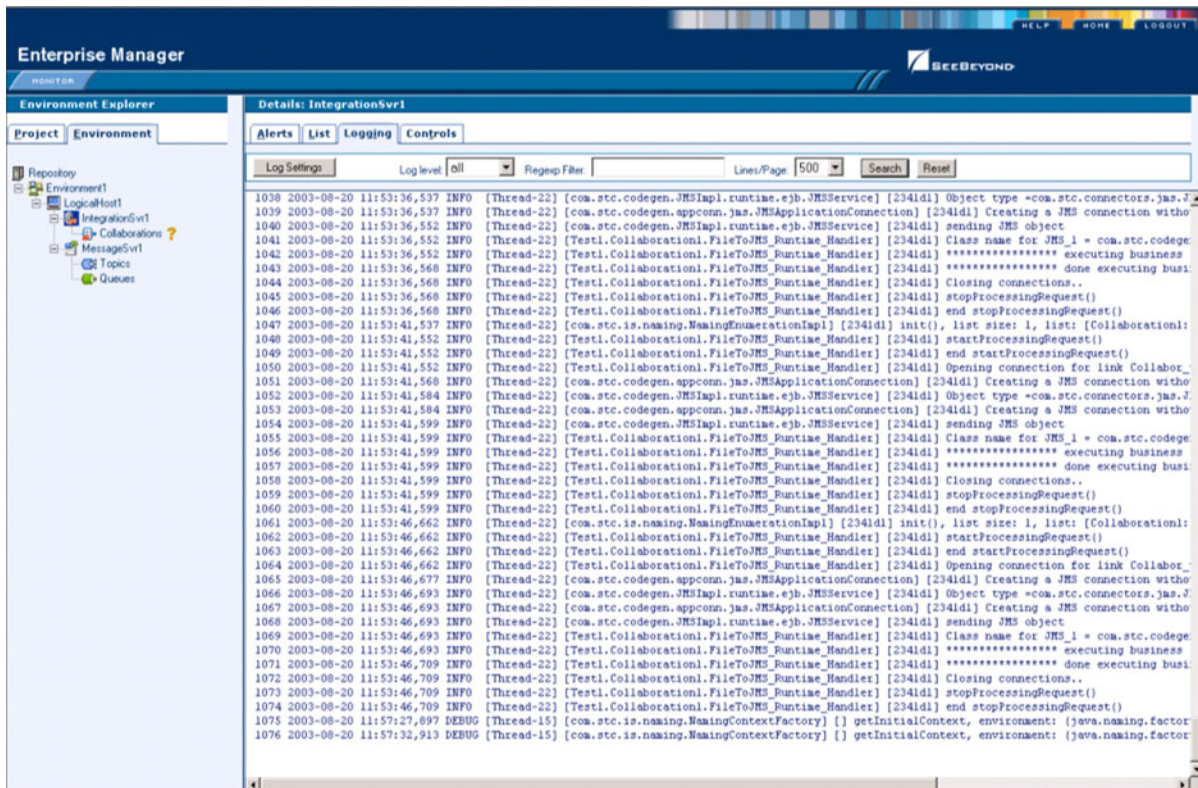
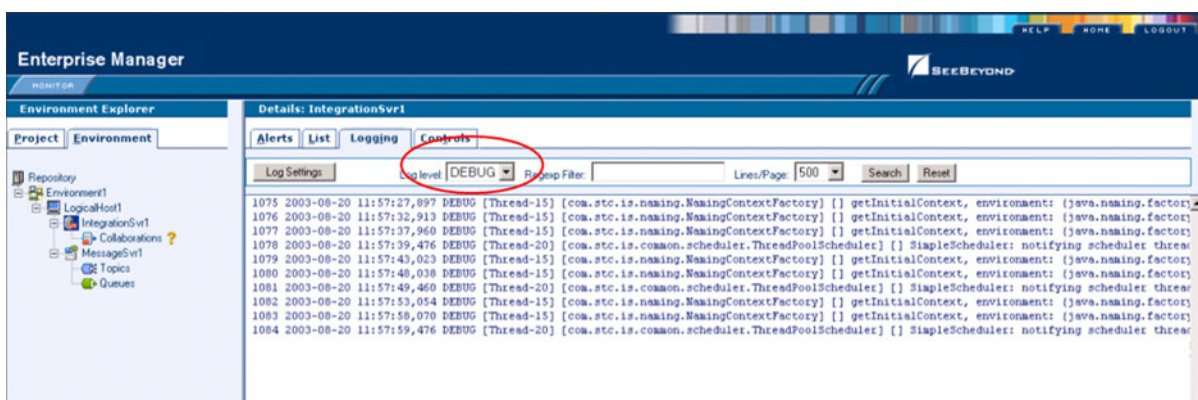


Figure 142 Integration Server Log Messages - Filtered



11.5.2 Collaboration Level

Select a Collaboration, click the **Log** tab in the upper Details panel, and click the **Logging** tab in the lower Details panel to display all log messages for the Collaboration (see Figure 143). This tab contains controls for viewing, sorting, and filtering of the log files on the server. By entering a keyword, you can search for a particular word in the log file (see Figure 144).

Figure 143 Collaboration Log File

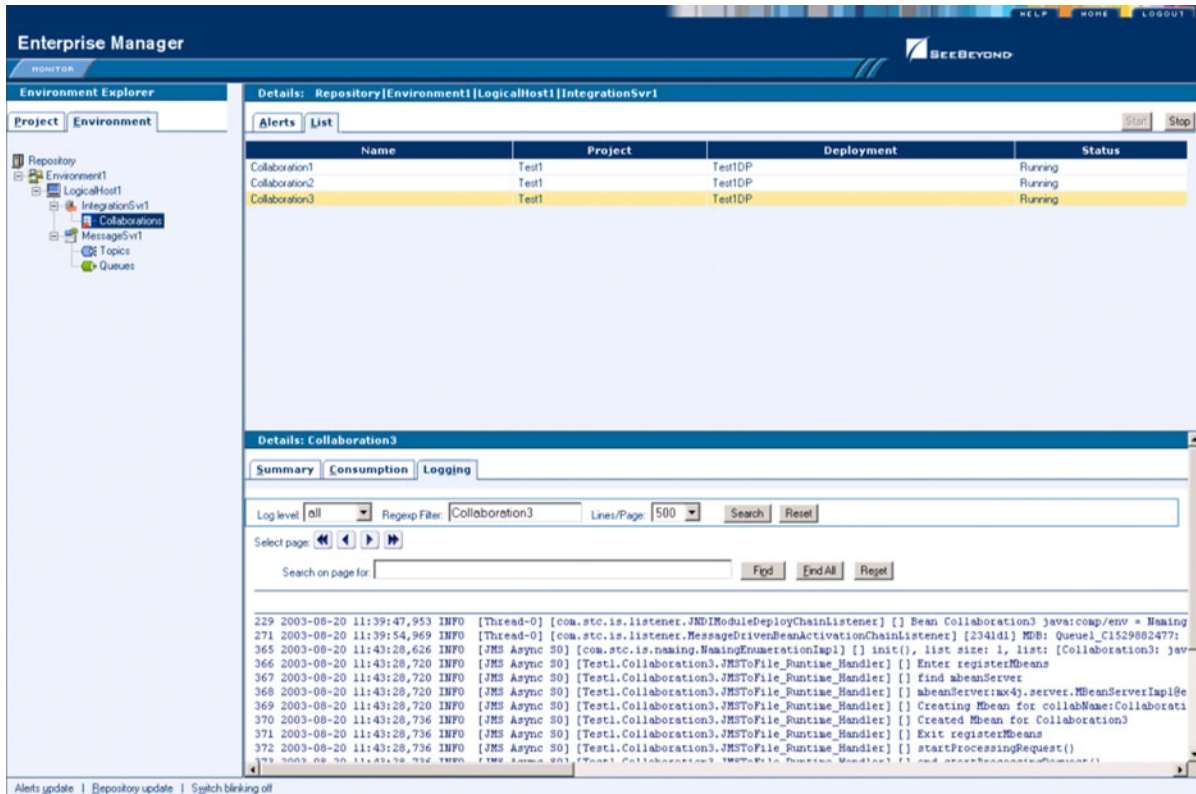
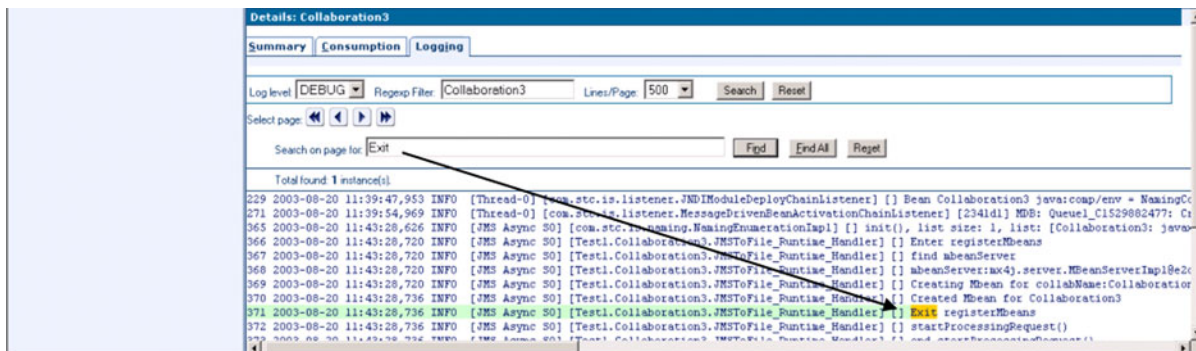


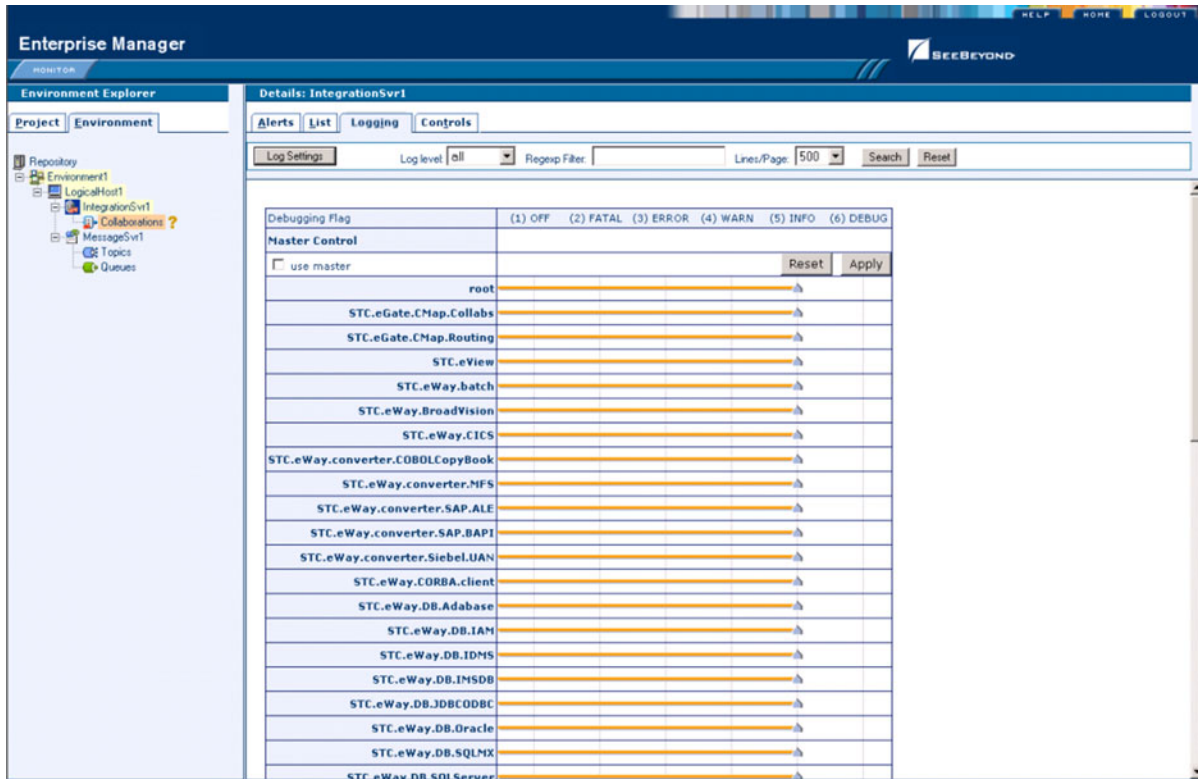
Figure 144 Search on Keyword



11.6 Setting Log Levels

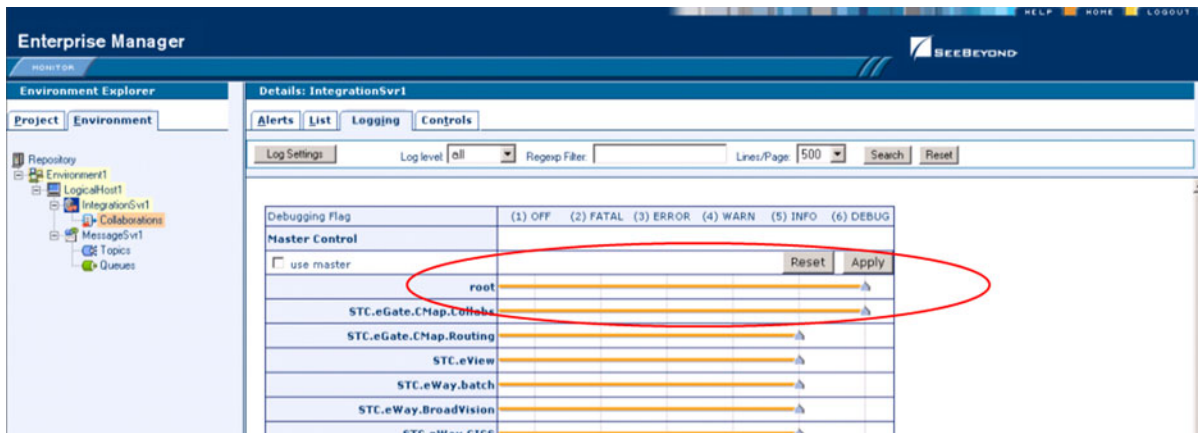
Select an Integration Server or Logical Host and click the **Log Settings** button to display the Logging Properties page (see Figure 145).

Figure 145 Logging Properties Page



To set a log level for a specific property, drag the line to the desired log level and click the **Apply** button. Figure 146 shows the first two properties reset to the **debug** level.

Figure 146 Resetting Log Levels



11.7 Indoubt Transaction Editing

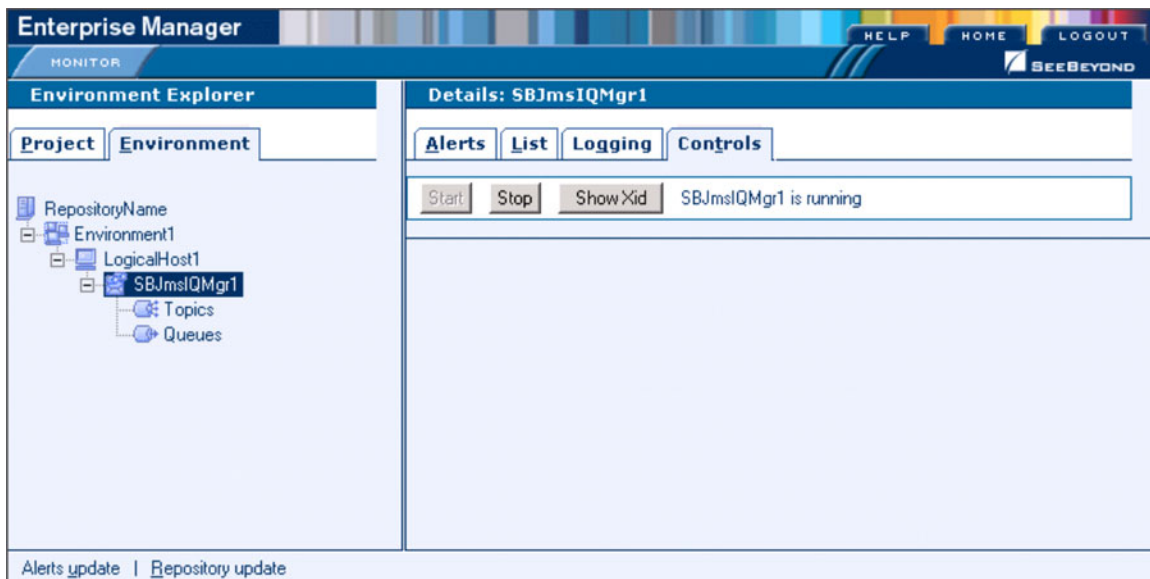
Occasionally, one of the Resource Managers (such as a database server or an external program) involved in a transaction will fail to commit. In that case, the transaction stays open until either the Resource Manager commits or rolls back, or the user intervenes.

The following feature is provided so that an Administrator can force those “indoubt” transactions to roll forward or backward. Typically, an external user will advise the Administrator of the problem, specifying the XID. The Administrator can then search for the indoubt transaction using this XID.

To force an Indoubt Transaction

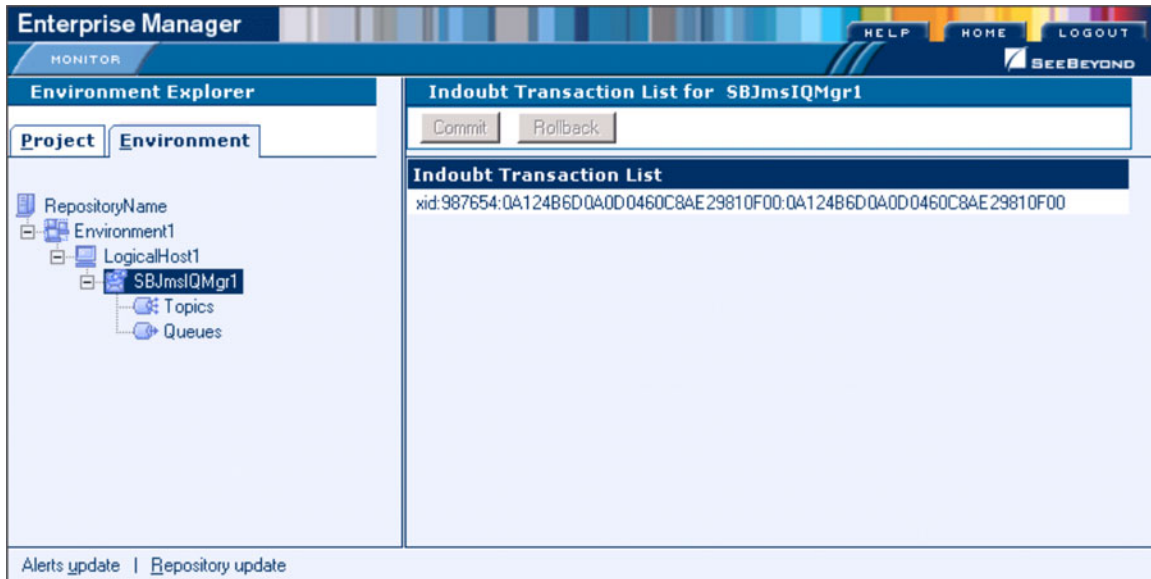
- 1 Click the **Controls** tab in the **Details** panel for the appropriate message server to display the interface shown in Figure 147.

Figure 147 Message Server Details - Controls Tab



- 2 Click the **ShowXid** button to display the Indoubt Transaction List shown in Figure 148.

Figure 148 Indoubt Transaction List



- 3 Select the transaction having the specified XID, as shown in Figure 149, and click either **Commit** or **Rollback**.

Figure 149 Indoubt Transaction List - Transaction Selected



Logging

This chapter provides information about eGate Integrator's logging features.

12.1 Logging

While a deployment profile is active and running, eGate Integrator automatically generates log messages for the Repository, Logical Host, and Enterprise Designer. The logging level determines what type of information is recorded, as the logs will only contain messages of the configured severity level and higher.

The log message files that are generated for the Repository, Logical Host, and Enterprise Designer, and their locations, are described in [Log Files and Locations](#) on page 176.

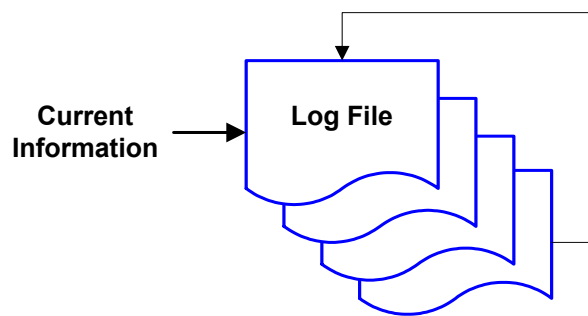
Viewing logs and setting log levels are accomplished in the Enterprise Manager, and are described in [Viewing Logs](#) on page 168 and [Setting Log Levels](#) on page 170.

12.2 eGate Logs

12.2.1 Log File System

While a deployment profile is active and running, eGate Integrator automatically generates log messages for the Repository, Logical Host, and Enterprise Designer. The log files constitute a recirculating stack (see Figure 150). You can specify both the file size and number of files in the stack for each Logical Host and Integration Server instance (the respective property names are `MaxFileSize` and `MaxBackupIndex`).

Figure 150 Recirculating Log File Stack



As soon as the maximum file size is reached in the currently active log file, a new log file is created. When the number of files in the stack reaches the specified limit, the oldest one is deleted when the new one is created. The effect is that the oldest one is emptied and moved to the top of the stack. A separate stack is maintained for each log file type.

Log files are initialized during the installation of a new logical host, so if you reinstall a logical host, all existing log files are deleted. If you would like to preserve log files (for example, on a weekly basis), you can copy the log files to a backup storage location.

12.2.2 Log File Management

Log files are managed from the Enterprise Manager; see:

- [Viewing Logs](#) on page 168
- [Setting Log Levels](#) on page 170

12.2.3 Log File Properties

Logical Host

Log file properties are set in the following properties file.

`ican50/logicalhost/logconfigs/LH/log4j.properties`

Table 27 Log Properties for Logical Hosts

Property	Default Value
log4j.appender.FILE	org.apache.log4j.RollingFileAppender
log4j.appender.FILE.File	C:/ican50/logicalhost/logs/stc_lh.log
log4j.appender.FILE.MaxFileSize	10MB
log4j.appender.FILE.MaxBackupIndex	10
log4j.appender.FILE.layout	org.apache.log4j.PatternLayout
log4j.appender.FILE.layout.ConversionPattern	%d{ISO8601} %-5p [%t] [%c] [%x] %m%n
log4j.rootCategory	INFO, FILE

***Note:** If it becomes necessary to increase space for Logical Host log files, you must shut down the Logical Host, change the **MaxFileSize** and/or **MaxBackupIndex** properties, and re-bootstrap the Logical Host.*

Integration Server

Log file properties are set in the following properties file.

`ican50/logicalhost/logconfigs/IS_integservername/log4j.properties`

Table 28 Log Properties for Integration Servers

Property	Default Value
log4j.appender.FILE	org.apache.log4j.RollingFileAppender
log4j.appender.FILE.File	C:/ican50/logicalhost/logs/stc_is_IntegrationSvr1.log
log4j.appender.FILE.MaxFileSize	10MB
log4j.appender.FILE.MaxBackupIndex	10
log4j.appender.FILE.layout	org.apache.log4j.PatternLayout
log4j.appender.FILE.layout.ConversionPattern	%d{ISO8601} %-5p [%t] [%c] [%x] %m%n
log4j.rootCategory	INFO, FILE

12.3 Log Files and Locations

Note: *The Repository Deployment Application log is related to all deployment actions spawned by invoking the **Apply** menu item from Enterprise Designer or by invoking the bootstrap script. If any errors occur during these invocations and the Enterprise Designer/bootstrap logs do not contain enough information to locate the problem, this log will contain the root cause of the problem—if it originated from the deployment application residing on the repository server.*

12.3.1 Repository

- Administration servlet log:

```
repository-root/repository/server/logs/hostname_admin_log.date.txt
```

- Examples servlet log:

```
repository-root/repository/server/logs/  
hostname_examples_log.<date>.txt
```

- Default repository and manifest servlet log:

```
repository-root/repository/server/logs/hostname_log.date.txt
```

- ♦ Init file:

```
repository-root/server/conf/log4j.properties.
```

- Repository installation log:

```
repository-root/repository/logs/install.log
```

- Master repository log:

```
repository-root/repository/logs/repository.log
```

- ♦ Init file:

```
repository-root/server/webapps/repositoryconfig.properties
```

- Deployment Application log:

```
repository-root/repository/logs/deployment-servlet.log
```

- ♦ Log4J init file:

```
repository-root/server/work/Standalone/localhost/deployment /WEB-  
INF/classes/deployment-servlet-log4j.properties.
```

- Enterprise monitor log.:

```
repository-root/repository/logs/install.log
```

- SeeBeyond Security Service (SSS) installation log:

```
repository-root/repository/logs/sss_install.log
```

- ♦ Log4J init file:

```
repository-root/server/SSS/logs/log4j-init.properties
```

- Velocity templates log (related to SSS).:

```
repository-root/repository/logs/velocity.log
```


12.3.2 Logical Host

- Master log file:

logical-host-root/stcma/stc_lh.log

- Init file:

logical-host-root/bootstrap/config/log4j.xml

12.3.3 Management Agent

- Master log file:

logical-host-root/stcma/stc_lh.log

- Deployment event log file:

logical-host-root/bootstrap/bin/deploy.log

- Deployment log file:

logical-host-root/stcma/deploy.log

- Velocity template log file:

logical-host-root/stcma/Velocity.log

- Init file:

logical-host-root/bootstrap/config/log4j.xml

12.3.4 Integration Server

- Master log file:

logical-host-root/stcis/logs/integ_server_instance_name/stc_is.log

- Init file:

*logical-host-root/test/stcis/integ-server-name/log4j-
config.properties*

12.3.5 Message Server

- Master log file:

logical-host-root/stcms/message-server-instance-name/stc_ms.log

- Init file:

*logical-host-root/stcms/message-server-instance-name/
stcms.default.Properties*

12.3.6 Enterprise Designer

- Master log file:

enterprise-designer-root/bin/stc.log

- Init file:

enterprise-designer-root/bin/log4j.properties

Glossary

Collaboration

(See Service and Collaboration Definition.)

Collaboration Definition

The encoding of business rules, in Java or XSLT format. Typically, the encoding consists of operations on OTDs (see **“OTD” on page 179**). Several Collaborations can have the same Collaboration Definition.

Connection

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

Connectivity Map

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

Constants

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

Deployment Profile

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

Derived Collaboration

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

eGate System

See **“Project”**.

Environment

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

eWay

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

External Application

A logical representation of an external application.

External System

A representation of an external application system.

ICAN Suite

The SeeBeyond Integrated Composite Application Network Suite, which is based on eGate Integrator.

Integration Server

Software platform that houses the business logic container used to run Collaborations. Provides transaction services, persistence, and external connectivity.

Link

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

Linked Message Destination

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

Logical Host

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

Management Agent

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

Message Destination

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

Message Server

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

OTD

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

Project

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

Queue

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

Repository

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

Schema Runtime Environment

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

Security Server

A standalone server that is the connection point to underlying eGate security environments.

Service

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

Subproject

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

Topic

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

XSLT

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

e*Gate 4.x Terms in eGate 5.0

Table 29 provides definitions for the terms that are new with eGate release 5.0, as well as equivalent terms from eGate release 4.x.

Table 29 eGate 5.0 Terms

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

Table 29 eGate 5.0 Terms (Continued)

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

Index

A

ACL properties 34–35, 40–41
 activate deployment profile 92
 admin 153
 analysis questions 118

C

Collaboration 178, 181
 derived 178
 Collaboration definition 178, 181
 commands
 Open File 83
 Refresh OTD 83
 Run Tester 83
 Save as New Name 83
 Save File 83
 Sort by Name 83
 Tester 83
 Toggle Reference Tab Panel 83
 component data 118
 connection 178, 181
 Connectivity Map 178, 181
 toolbar 53
 Connectivity Map Editor 52
 constants 178
 Control Broker 181
 conventions, writing in document 16
 customizer 58

D

deactivate deployment profile 92
 deploy 181
 Deployment 181
 Deployment Profile 91, 178, 181
 components 92
 deployment profile
 activate 92
 deactivate 92
 map variables 92
 derived Collaboration 178

E

e*Way 181
 e*Way Connection 181
 e*Way Connection Configuration 181
 Editor
 Connectivity Map 52
 eGate system 178
 enter file name to export 126
 Enterprise Designer 181
 enterprise explorer 31
 menu bar 28
 object type definition 69
 enterprise explorer
 environments view 36
 project explorer view 31
 Enterprise Manager 181
 Administration 153
 Documentation 157
 Downloads 156
 Home 149
 Home tab 149
 login 147
 monitor 149
 Enterprise Monitor 181
 Environment 178, 181
 components 36
 Environments 22
 ETD 181
 Event Type Definition 181
 eWay 178, 181
 eWay Configuration 181
 external
 application 179, 181
 system 179, 181

I

ICAN Suite 179
 impact analyzer
 command buttons 119
 overview 118
 printing 120
 toolbar button 30
 using 119
 indoubt transaction 171
 Integration Server 179, 181

J

JMS
 connection 181
 e*Way Connection 181
 IQ Manager 181

Index

queue 181
topic 182

L

link 179, 181
linked message destination 179, 181
Logical Host 179, 181
 Starting as a Windows Service 113
 Starting Manually on Windows 114
 Starting on an HP NonStop Server 116
 Starting on Linux 115
 Starting on UNIX 115
 Startup Configuration File 108
 Startup Parameters (Windows) 107

M

Management Agent 179
map variables 92
menu bar 28
message
 destination 179, 181
 server 179, 181
Monitor Server 149, 159

N

network view 181

O

Object Type Definition 179, 181
 wizard 75, 79
Open File command 83
OTD 179, 181

P

Participating Host 181
Process Manager 181
Project 179, 181
 project explorer view 31

Q

queue 180–181

R

Refresh OTD command 83
Registry 181
Repository 22, 180–181

roles 131
Run Tester command 83

S

Save as New Name command 83
Save File command 83
Schema 181
Schema Runtime Environment 180
security 106
Security Server 180
select file to import 128
Sort by Name command 83
SRE 180
subproject 180–181
supporting documents 17

T

Tester command 83
Toggle Reference Tab Panel command 83
topic 180–182

U

user management 132

V

version control 121–123

X

XSLT 180, 182