SeeBeyond ICAN Suite

eGate Integrator for eInsight Enterprise Service Bus User's Guide

Release 5.0.1

SeeBeyond Proprietary and Confidential

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 This work is protected as an unpublished work under the copyright laws.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 20031121165529.

Contents

List of Figures 9

List of Tables 14

Chapter 1

Introduction 16

Purpose and Scope 16
Intended Audience 16
Organization of Information 16
Writing Conventions 17
Supporting Documents 18
Online Documents 18
The SeeBeyond Web Site 18

Chapter 2

System Overview 19

Introduction 19
Integration Model 20
System Architecture 22
Repository 23
Environments 23
User Interfaces 24
Enterprise Designer 24
Enterprise Manager 26

Chapter 3

Enterprise Manager 27

Overview 27
Installing and Updating eGate 27

Monitoring and Managing eGate 27
Starting Enterprise Manager 28
The Enterprise Manager Interface 29
Home 30
Documentation 30
The Enterprise Monitor 31
Viewing Alerts 33
Viewing Logs 36
Integration Server Level 36
Collaboration Level 37
Setting Log Levels 38
Editing XA Transactions 39

Chapter 4
Enterprise Designer 41
Overview 41
Starting Enterprise Designer 43
Interface Features 44
Menus 44
File Menu 44
Tools Menu 44
View Menu 45
Window Menu 45
Help Menu 46
Toolbar 46
Browser Buttons 47
Enterprise Explorer 48
Project Explorer 48
Environment Explorer 49
Enterprise Designer Editors 50
Connectivity Map Editor 51
OTD Editor 52
Environment Editor 53
Deployment Editor 54
Additional Tools and Features 55
Impact Analyzer 55
Version Control 57
Checking a Component In 57
Checking a Component Out 58
Viewing a Component's Version History 59
Repository Backup and Restoration 60
Backing Up a Repository 60
Restoring a Repository 60
Project Export and Import 62
Exporting a Project 62
Exporting a Project Using Enterprise Designer 62

Exporting a Project Using the Command Line 63
Importing a Project 64
Importing a Project Using Enterprise Designer 64
Importing a Project Using the Command Line 65

Chapter 5
eGate Projects 67
Overview 67
Project Components 67
The Project Explorer 68
Project Explorer Icons 69
Context Menus 70
Repository Menu 70
Project Menu 71
Connectivity Map Menu 72
Using the Connectivity Map Editor 73
Message Destinations 75
Topics 75
Queues 75
External Applications 76
Schedulers 77
Component Connections 78
Configuring a Connection 79
Modifying a Configuration Property 81
Defining Constants and Variables 82

Chapter 6
Web Services 85
Overview 85
SeeBeyond Web Services 86
UDDI Repository 87
Building a Web Client 89
Building a Web Server 97

Chapter 7
Object Type Definitions 104
Overview 104
OTD Types 104

Externally-Defined OTDs 104
Building OTDs 104
Using the OTD Wizard 105
Externally-Defined OTDs 107
Using the DTD Wizard 107
Using the WSDL Wizard 111
WSDL OTD Structure 114
WSDL Operation Elements 115
Using the XSD Wizard 116
Using the OTD Editor 120
Node Management 122
Using the OTD Tester 123

Chapter 8
Environments 126
Overview 126
Environment Explorer 126
Environment Editor 127
Environment Explorer Icons 128
Context Menus 129
Repository Menu 129
Environment Menu 130
Logical Host Menu 131
Defining Environmental Constants 133
Logical Hosts 135
Management Agent 136
Message Servers 136
Integration Servers 137
Configuring the Logical Host 138
Modifying the Logical Host Properties File 138
Installing the Logical Host as a Windows Service 141
Configuring the Base Port Number 143
Starting the Logical Host 144
Starting the Logical Host on a Windows System 145
Starting the Logical Host on a UNIX System 145
Starting the Logical Host on a Red Hat Linux System 145
Stopping the Logical Host 146

Chapter 9
Project Deployment 147
Deployment Profiles 147

Using the Deployment Editor 148
Creating a Deployment Profile 149
Activating and Deactivating Deployment Profiles 152
Mapping Variables 154
Deploying Projects to Third-Party Servers 155
BEA WebLogic 155
IBM WebSphere 158

Chapter 10
ICAN Security Features 161
Overview 161
Multiple Environments 162
User Management 163
Configuration User Management 163
Roles 163
Adding and Deleting Repository Users 163
Adding and Deleting User Roles 167
Creating User Roles 172
Using LDAP 176
Configuring LDAP Servers for Configuration User Management 178
Configuring the Active Directory Server 178
Configuring the SunONE Server 184
Configuring the ICAN Repository to Use LDAP 186
Configuring the ICAN Suite for ADS 187
Configuring the ICAN Suite for SunONE 187
Environment User Management 188
Creating and Configuring Users 188
Configuring for LDAP Servers in Environment User Management 189
ACL Management 190
ICAN Component Security 193
Message Server Security 193
JMS Client Security 193
Using SSL/HTTPS in ICAN 195
Overview 195
Certificates and Keys 195
The Keytool Utility 195
Installation and Configuration 196

Chapter 11
Logging 199
Overview 199
eGate Logs 200

Log File System 200
Log File Management 200
Logging Model 201
Loggers 201
Appenders 201
Layouts 201
Basic Log Files and Locations 202
Repository 202
ESR Installer Logs 203
Enterprise Designer 204
Enterprise Monitor 205
Run-Time Log Files and Locations 206
Logical Host 206
Integration Servers 207
Message Servers 207
Glossary 208
e*Gate 4.x Terms in eGate 5.0 214
Index 216

List of Figures

Figure 1	SeeBeyond ICAN Suite	19
Figure 2	eGate Integrator	20
Figure 3	eGate Integrator Implementation Model	21
Figure 4	Typical eGate Integrator System	22
Figure 5	Enterprise Designer	24
Figure 6	Connectivity Map Editor	25
Figure 7	SeeBeyond Enterprise Manager Login	26
Figure 8	Enterprise Manager Login	28
Figure 9	Enterprise Manager GUI	29
Figure 10	Monitor Launch Window	30
Figure 11	Documentation Tab	30
Figure 12	Monitor Interface - Initial	31
Figure 13	Monitor Interface - Environment	31
Figure 14	Collaboration Alerts	33
Figure 15	Alert Status	33
Figure 16	Alert Details	34
Figure 17	Integration Server Log Messages	36
Figure 18	Integration Server Log Messages - Filtered	36
Figure 19	Collaboration Log File	37
Figure 20	Search on Keyword	37
Figure 21	Logging Properties Page	38
Figure 22	Resetting Log Levels	38
Figure 23	Message Server Details - Controls Tab	39
Figure 24	Indoubt Transaction List	40
Figure 25	Indoubt Transaction List - Transaction Selected	40
Figure 26	SeeBeyond Enterprise Designer	41
Figure 27	Login Dialog Box	43
Figure 28	Enterprise Explorer: Project Explorer View	48
Figure 29	Enterprise Explorer: Environment Explorer View	49
Figure 30	Connectivity Map Editor	51
Figure 31	OTD Editor	52
Figure 32	Environment Editor	53

Figure 33	Deployment Editor	54
Figure 34	Impact Analyzer Dialog Box	55
Figure 35	Version Control - Check In Dialog Box	57
Figure 36	Version Control - Check Out Dialog Box	58
Figure 37	Version Control - History Dialog Box	59
Figure 38	Enter File Name Dialog Box	62
Figure 39	Message Dialog Box	63
Figure 40	Select File Dialog Box	64
Figure 41	File Destination Dialog Box	65
Figure 42	Message Dialog Box	65
Figure 43	Project Explorer	68
Figure 44	Repository Menu	70
Figure 45	Project Menu	71
Figure 46	Connectivity Map Menu	72
Figure 47	Connectivity Map Window	73
Figure 48	Linking JMS Topics	73
Figure 49	External Application Drop-Down Menu	76
Figure 50	Selected External Applications in Toolbar	76
Figure 51	Scheduler Component	77
Figure 52	Connection Icons in a Connectivity Map	78
Figure 53	Default Configuration Dialog Box	79
Figure 54	Editing a Default Configuration Field	81
Figure 55	Project Variable Creation	82
Figure 56	Project Constant Creation	83
Figure 57	Variables and Constants Object Group	83
Figure 58	Connector Properties	84
Figure 59	SeeBeyond UDDI Repository	87
Figure 60	Example Web Service WSDL File	87
Figure 61	Microsoft Visual Studio Example	88
Figure 62	File Destination Dialog Box	89
Figure 63	Select WSDL Wizard	90
Figure 64	Select File Location	91
Figure 65	Select WSDL File	91
Figure 66	Select External Server	92
Figure 67	eInsight Business Process - Receive Rule	93
Figure 68	eInsight Business Process - Write Rule	94
Figure 69	Connectivity Map	95
Figure 70	Project Deployment	96

Figure 71	File Destination Dialog Box	97
Figure 72	Select WSDL Wizard	98
Figure 73	Select File Location	99
Figure 74	Select WSDL File	99
Figure 75	Select External Client	100
Figure 76	eInsight Business Process	101
Figure 77	Connectivity Map	102
Figure 78	Project Deployment	103
Figure 79	OTD Wizard Selection Dialog	105
Figure 80	OTD Wizard Selection: DTD Wizard	107
Figure 81	Select DTD File(s) Dialog Box	108
Figure 82	Select Document Elements Dialog Box	109
Figure 83	Select OTD Options Dialog Box	110
Figure 84	OTD Wizard Selection: WSDL Wizard	111
Figure 85	WSDL Wizard: Select WSDL Location	112
Figure 86	WSDL Wizard: Select WSDL File	113
Figure 87	WSDL Wizard: Select OTD Options	114
Figure 88	OTD Wizard Selection: XSD Wizard	116
Figure 89	XSD Wizard: Select XSD File(s)	117
Figure 90	Select Document Elements Dialog Box	118
Figure 91	Select OTD Options Dialog Box	119
Figure 92	OTD Editor	120
Figure 93	OTD Tester	123
Figure 94	Test Panel Data Display	123
Figure 95	Select Data File	124
Figure 96	Object Elements and Values	124
Figure 97	Data Display: Refresh Icon	125
Figure 98	Status Data Display	125
Figure 99	Enterprise Explorer: Environment Explorer View	126
Figure 100	Environment Editor	127
Figure 101	Repository Menu	129
Figure 102	Environment Menu	130
Figure 103	Logical Host Menu	131
Figure 104	Environmental Constants Editor	133
Figure 105	Logical Hosts	135
Figure 106	Integration Server (J2EE Compatible)	137
Figure 107	Example logical-host.properties File	138
Figure 108	Install as Service Script	141

Figure 109 Windows Services List 141
Figure 110 Uninstall as Service Script 142
Figure 111 Logical Host Properties Dialog Box 143
Figure 112 eGate Integrator Implementation Model 147
Figure 113 Deployment Editor Window 148
Figure 114 Example Deployment Profile (1) 149
Figure 115 Example Deployment Profile (2) 150
Figure 116 Example Deployment Profile (3) 151
Figure 117 Activate Dialog Box 152
Figure 118 Logical Host Context Menu - Apply 152
Figure 119 Activate Dialog Box 153
Figure 120 Deployment Profile Mappings 154
Figure 121 Project Variable Value Entry 154
Figure 122 WebLogic Deployment (1) 155
Figure 123 WebLogic Deployment (2) 156
Figure 124 WebLogic Deployment Verification 157
Figure 125 WebSphere Deployment (1) 158
Figure 126 WebSphere Deployment (2) 159
Figure 127 WebSphere Deployment Verification 160
Figure 128 User Management Dialog Box (1) 164
Figure 129 User Management - Add User 164
Figure 130 User Management Dialog Box (2) 165
Figure 131 User Management Dialog Box 166
Figure 132 Error Warning Box 166
Figure 133 User Management Dialog Box 167
Figure 134 User Management - Add Role (1) 168
Figure 135 Add Role Dialog Box 168
Figure 136 User Management - Add Role (2) 169
Figure 137 User Management Dialog Box 169
Figure 138 User Management Dialog Box 170
Figure 139 User Management - Delete Role 171
Figure 140 Error Warning Box 171
Figure 141 User Management Users List 172
Figure 142 User Management - Create Role (1) 173
Figure 143 Add Role Dialog Box (1) 173
Figure 144 Role Dialog Box 174
Figure 145 Add Role Dialog Box (2) 174
Figure 146 User Management - Create Role (2) 175

Figure 147 User Management Users List 175
Figure 148 LDAP Directory Tree (Traditional Naming) 177
Figure 149 LDAP Directory Tree (Internet Naming) 177
Figure 150 Active Directory Users and Computers Window 179
Figure 151 Active Directory - Create Organizational Unit 180
Figure 152 Active Directory - Create Roles Groups 181
Figure 153 Active Directory - ICANRoles Directory 182
Figure 154 Active Directory - Select Administrator 183
Figure 155 Sun ONE - Create New Role 184
Figure 156 Sun ONE - Roles 185
Figure 157 Environment Context Menu 188
Figure 158 Security Configuration Template Properties Dialog Box 189
Figure 159 ACL Management Dialog Box (1) 191
Figure 160 ACL Add Users Dialog Box 191
Figure 161 ACL Management Dialog Box (2) 192
Figure 162 ACL Management Dialog Box (3) 192
Figure 163 JMS IQ Manager Properties 193
Figure 164 JMS Client Security Properties 194
Figure 165 Recirculating Log File Stack 200

List of Tables

Table 1	Writing Conventions	17
Table 2	Enterprise Manager - Tabs	29
Table 3	Enterprise Manager - Tabs	29
Table 4	Monitor Interface - Details Tabs	31
Table 5	Basic Alert Types	34
Table 6	File Menu Options	44
Table 7	Tools Menu Options	44
Table 8	View Menu Options	45
Table 9	Window Menu Options	45
Table 10	Help Menu Options	46
Table 11	Enterprise Designer Toolbar Icons	46
Table 12	Browser Buttons	47
Table 13	Impact Analyzer Command Buttons	56
Table 14	Project Icons	69
Table 15	Repository Menu Options	70
Table 16	Project Menu Options	71
Table 17	Connectivity Map Menu Options	72
Table 18	Connectivity Map Toolbar Icons	74
Table 19	Configuration Dialog Box Toolbar Buttons	80
Table 20	OTD Wizard Navigation Buttons	106
Table 21	OTD Editor Toolbar Icons	121
Table 22	Environment Icons	128
Table 23	Repository Menu Options	129
Table 24	Environment Menu Options	130
Table 25	Logical Host Menu Options	131
Table 26	Environmental Constants Editor Icons	134
Table 27	User-Modifiable Logical Host Properties	140
Table 28	Startup Command Arguments	144
Table 29	Deployment Toolbar Buttons	148
Table 30	Realm Element Attributes	186
Table 31	Logging Levels	201
Table 32	Properties for the Master Repository Log	202

Table 33	Properties for the UDDI Repository Log	203
Table 34	Properties for the ESR Installer Log	203
Table 35	Properties for the Enterprise Designer Master Log	204
Table 36	Properties for Enterprise Monitor Log	205
Table 37	Properties for the Logical Host Log	206
Table 38	Properties for the Integration Server Logs	207
Table 39	Properties for Integration Server Logs	207
Table 40	eGate 5.0 Terms	214

Chapter 1

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 operational 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 and UNIX) 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 Manager"** provides a detailed overview of the Enterprise Manager, including its structure and operation.

- **Chapter 4 "Enterprise Designer"** provides a detailed overview of the Enterprise Designer, including its structure and operation.
- **Chapter 5 "eGate Projects"** explains how to create a Connectivity Map and use the Configuration Editor to modify eWay and JMS connections between Connectivity Map components.
- **Chapter 6 "Web Services"** describes how to use the Web Services features of eGate Integrator.
- **Chapter 7 "Object Type Definitions"** describes how to create Object Type Definitions (OTDs).
- **Chapter 8"Environments"**explains how to create and populate eGate Environments, and how to configure and start Logical Hosts.
- **Chapter 9 "Project Deployment"** explains how to create and activate deployment profiles.
- **Chapter 10"ICAN Security Features"**discussed the various security features in the ICAN Suite.
- **Chapter 11 "Logging"** provides troubleshooting tips and describes the eGate logs

In addition, refer to the **Glossary** on page 208 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	`java -jar EnterpriseDesigner.jar`
Hypertext Links	Blue text	For more information, see **"Writing Conventions" on page 17**.
Notes	***Bold Italic*** text	***Note:*** *If a toolbar button is dimmed, you cannot use it with the selected component.*

1.5 Supporting Documents

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

- *eGate Integrator JMS Reference Guide*
- *eGate Integrator Release Notes*
- *eGate Integrator Tutorial*
- *eInsight Enterprise Service Bus User's Guide*
- *SeeBeyond ICAN Suite Deployment Guide*
- *SeeBeyond ICAN Suite Installation 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

Chapter 2

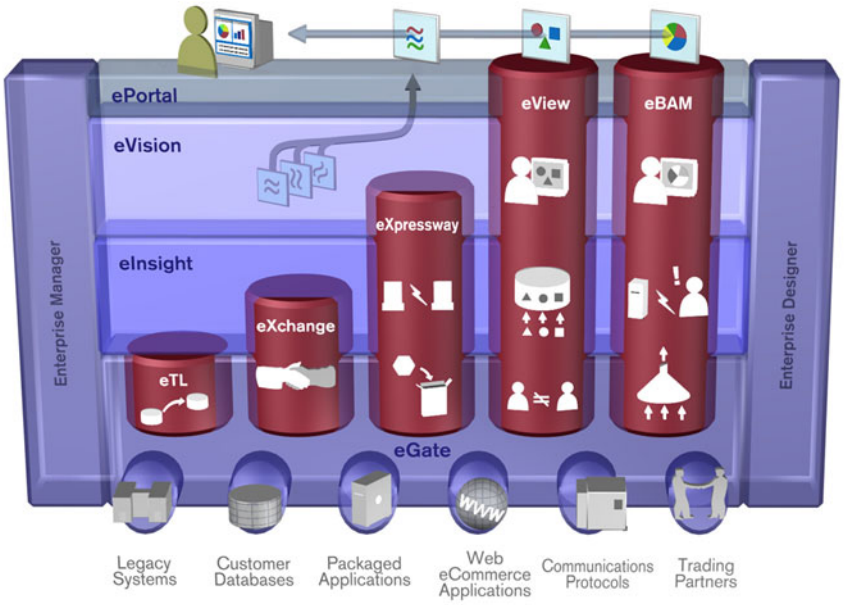
System Overview

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.

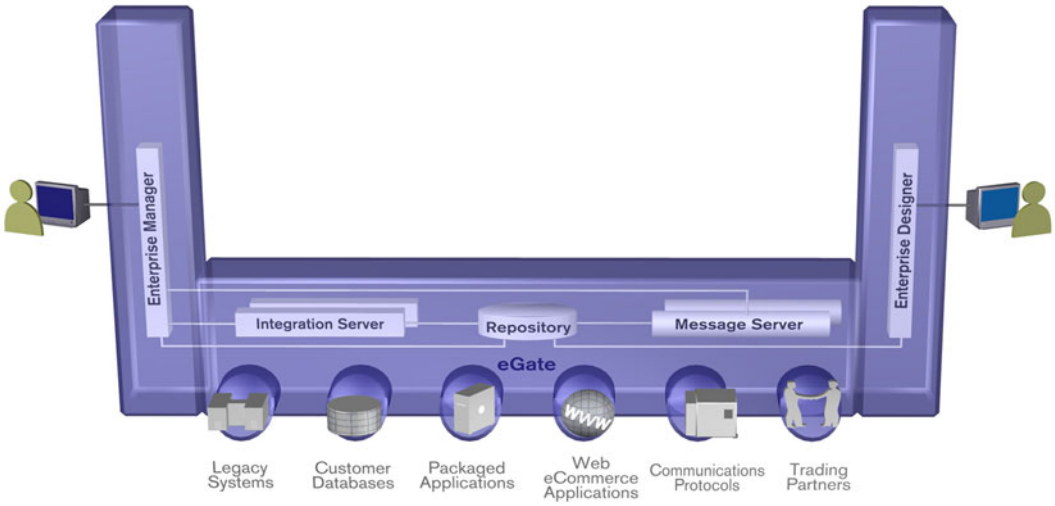
Figure 1 SeeBeyond ICAN Suite



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 26 and **Enterprise Designer** on page 24.

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.

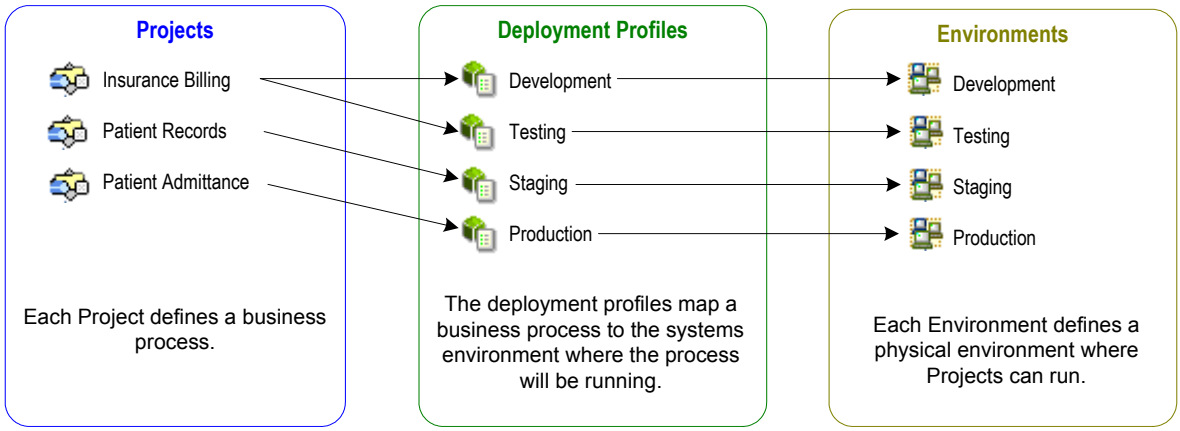
Projects are run within Logical Hosts, which are individual, runtime instances of eGate Integrator. Logical Hosts are defined within Environments, which represent the physical resources required to implement the Project. Projects are mapped to the individual Environments by means of Deployment Profiles,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

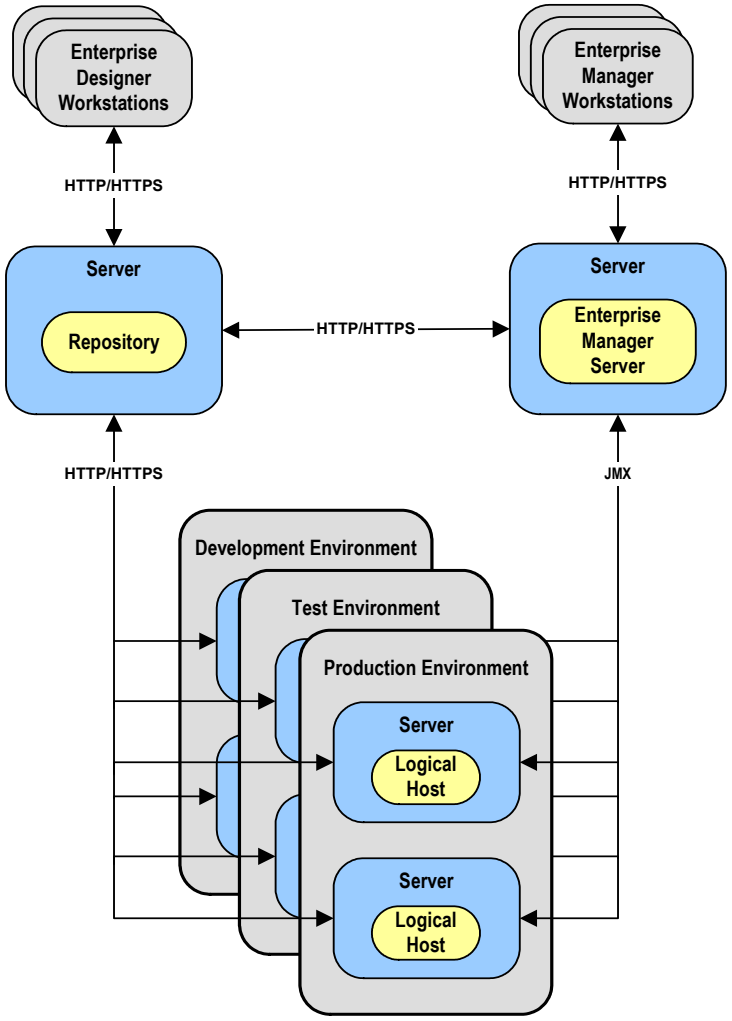
1. **Design your Project.**
2. **Define your Environments.**
3. **Create your Deployment Profiles.**
4. **Deploy the eGate Project.**

These implementation steps are all accomplished using the eGate Enterprise Designer, which is introduced in **Enterprise Designer** on page 24 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** such as the SeeBeyond JMS IQ Manager, 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.

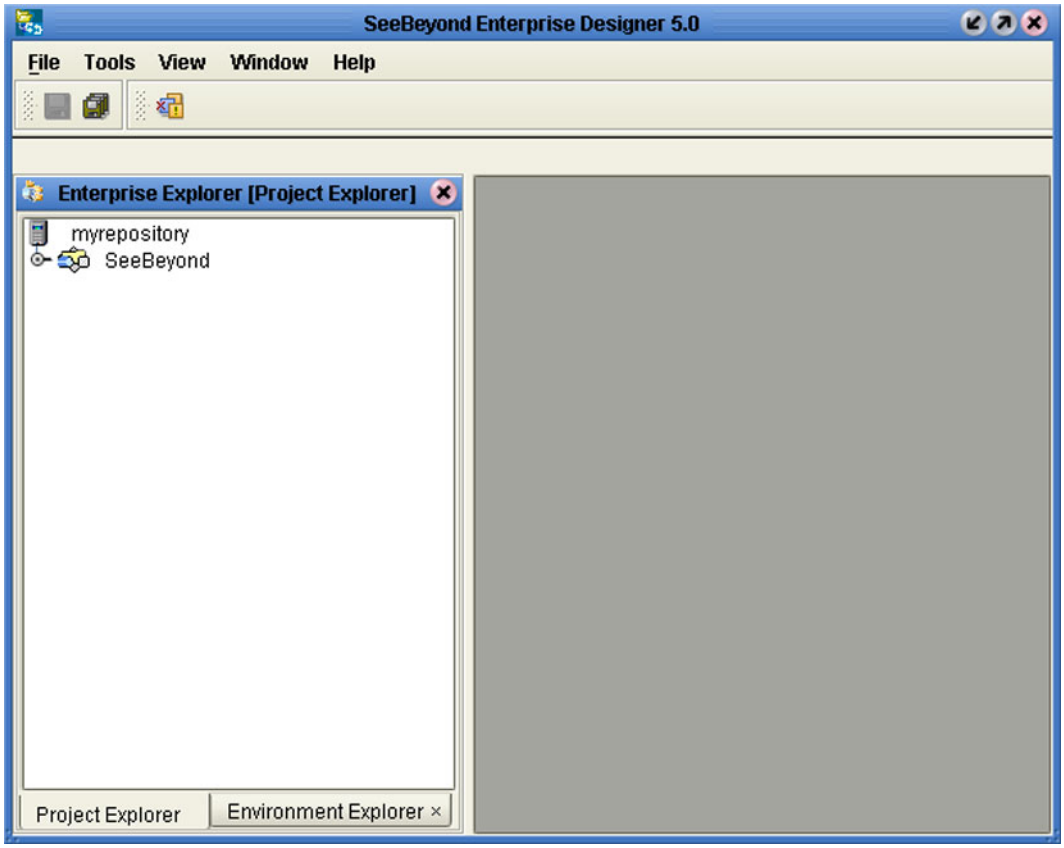
Note again that there is no requirement for the components shown in Figure 4 to run on separate systems; all could run on a single system, provided that resources (CPU, memory, and disk) are sufficient to support the concurrent usage.

2.4 User Interfaces

2.4.1 Enterprise Designer

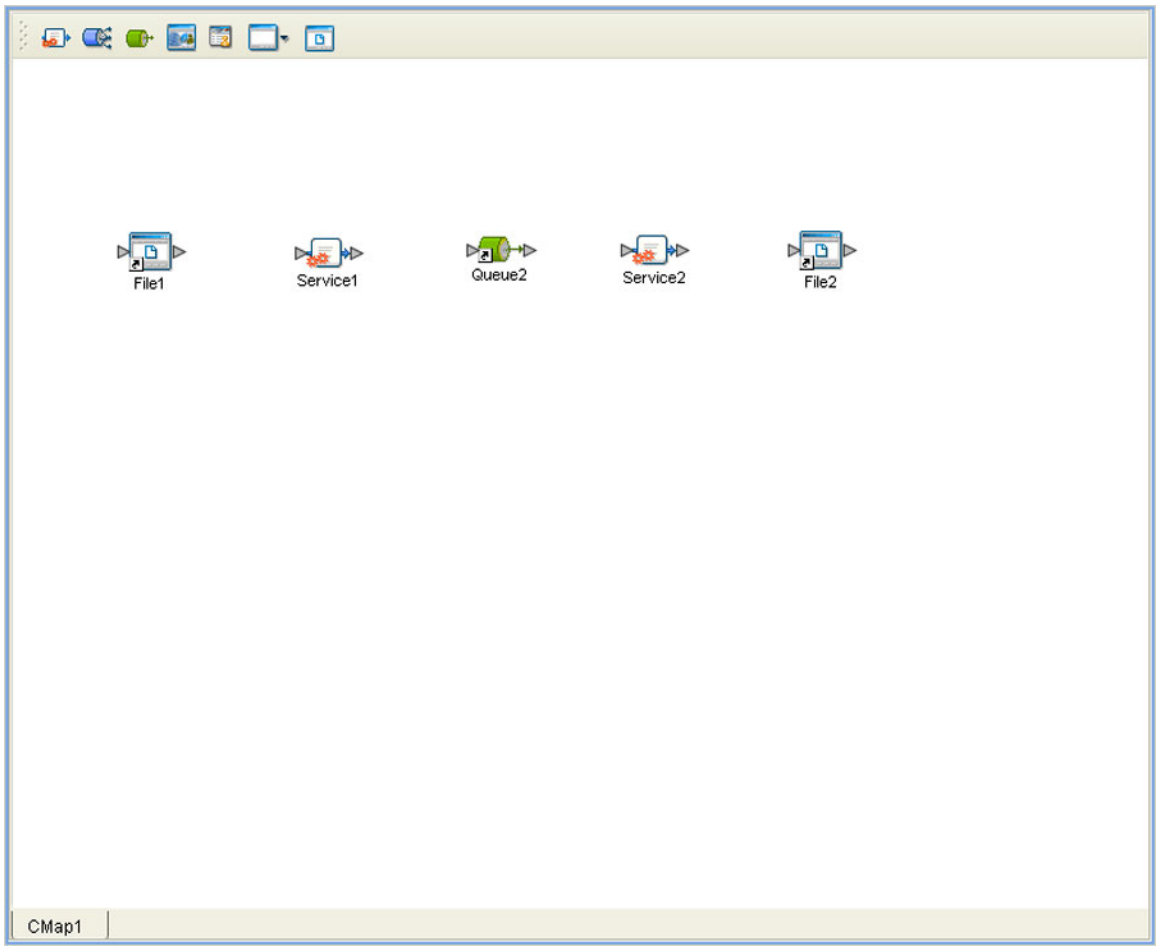
The Enterprise Designer graphical user interface (GUI) is used to create and configure the logical components and physical resources of an eGate Project. Through this GUI (see Figure 5), you can develop Projects to process and route data through an eGate Integrator system.

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 **eGate Projects** on page 67. 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 eInsight and 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 41* .

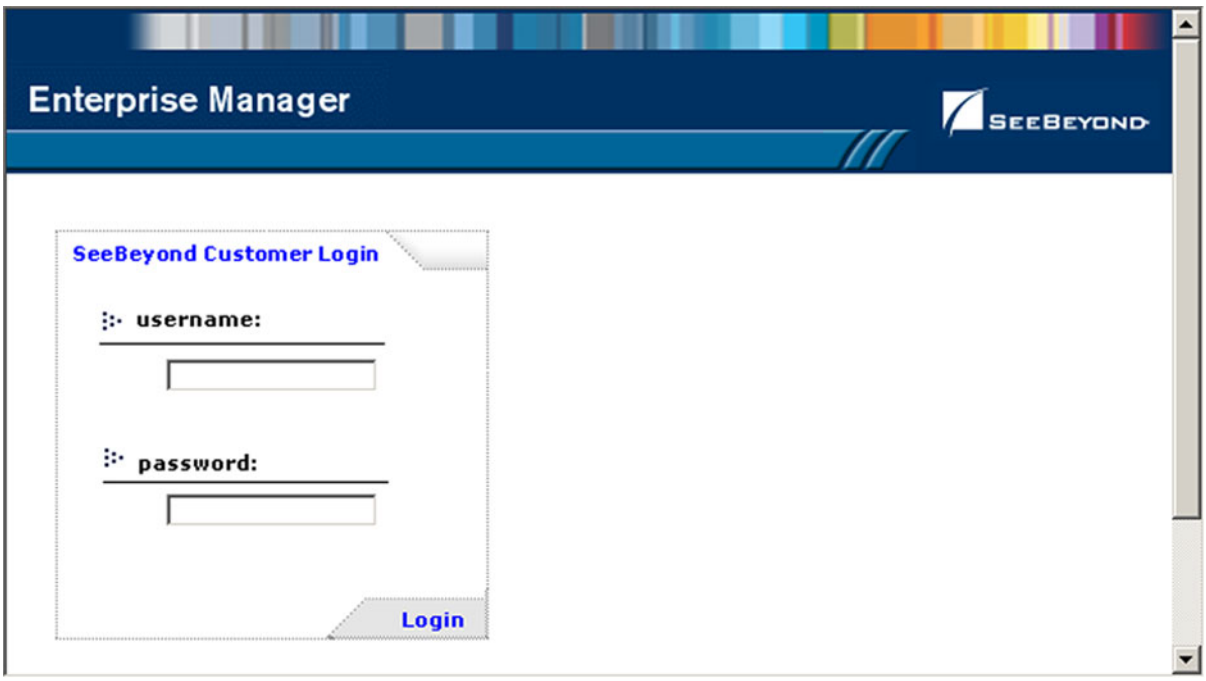
2.4.2 Enterprise Manager

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



For more information on the Enterprise Manager, see **Enterprise Manager** *on page 27*.

For more information on Web Services capability, see **Web Services** on page 85.

Chapter 3

Enterprise Manager

This chapter provides an introduction to the ICAN Suite Enterprise Manager.

3.1 Overview

Enterprise Manager is a Web-based interface with which you can install and update eGate Integrator, and monitor and manage deployed eGate components.

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

3.1.1 Installing and Updating eGate

eGate Integrator components 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. For information on installing and updating eGate components, see the *eGate Integrator Installation Guide*.

3.1.2 Monitoring and Managing eGate

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

- **The Enterprise Monitor** on page 31 describes features of the Monitor interface itself.
- **Viewing Alerts** on page 33 describes how to view and set the status of Alerts.
- **Viewing Logs** on page 36 briefly describes how to view, sort, search, and filter messages in the log files.
- **Setting Log Levels** on page 38 describes how to set logging levels in Enterprise Manager.

3.2 Starting Enterprise Manager

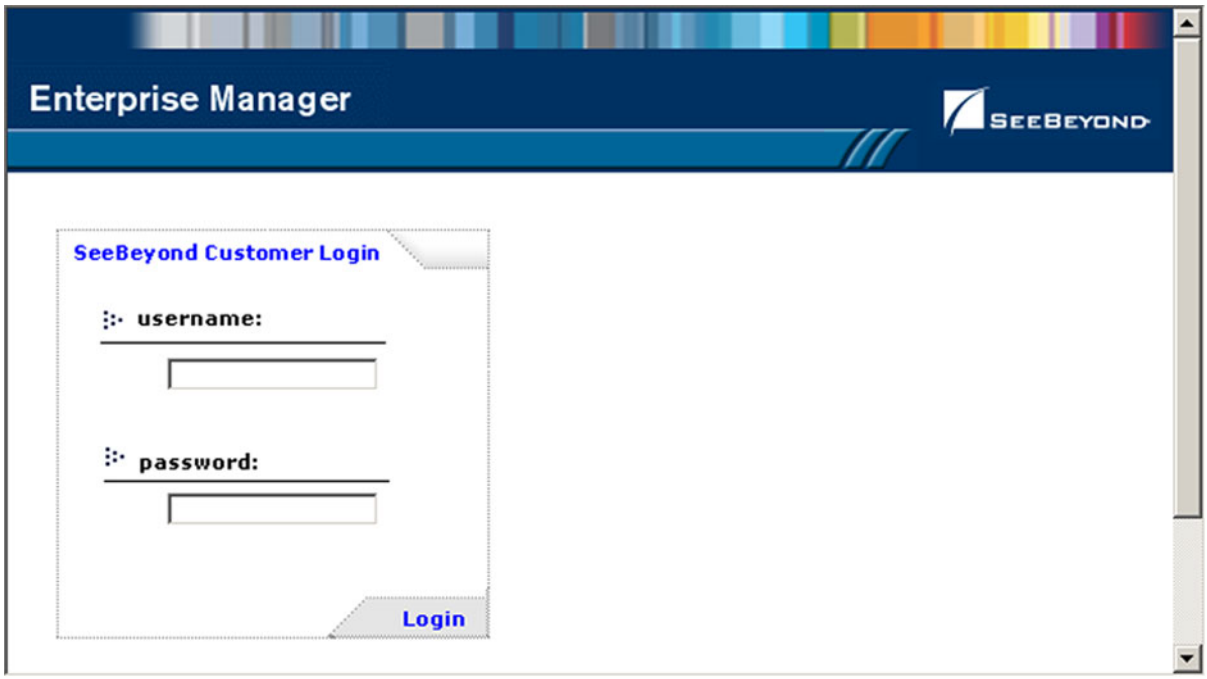
To start the Enterprise Manager

1. Launch Internet Explorer.
2. Enter **http://*hostname:portnumber*** in the **Address** box to display the SeeBeyond Customer Login window shown in Figure 8.

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. See the SeeBeyond ICAN Suite Installation Guide.*

Important: *The TCP/IP host name must be alphanumeric.*

Figure 8 Enterprise Manager Login

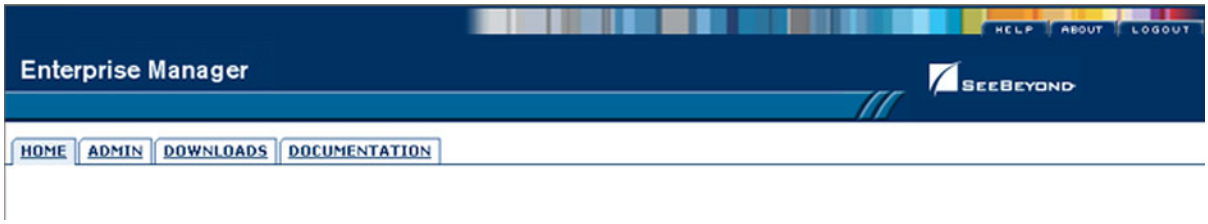


3. Enter your login ID and password in the **Username** and **Password** boxes and click **Login**.

3.3 The Enterprise Manager Interface

Once you have logged in, you see the full Enterprise Manager user interface (see Figure 9).

Figure 9 Enterprise Manager GUI



The Enterprise Manager is organized into four pages represented by tabs, as described in the following table.

Table 2 Enterprise Manager - Tabs

Tab	Function
Home	The Home tab is used for accessing the eGate Monitor, which is the main subject of this chapter. See **Home** on page 30.
Administration	The Administration tab is used in installing and updating ICAN components. See the *SeeBeyond ICAN Suite Installation Guide* for information.
Downloads	The Downloads tab is used in installing and updating ICAN components. See the *SeeBeyond ICAN Suite Installation Guide* for information.
Documentation	The Documentation tab is used for accessing ICAN Suite documentation. See **Documentation** on page 30.

There are also three small tabs in the upper-right corner of the Enterprise Manager, which are described in the following table.

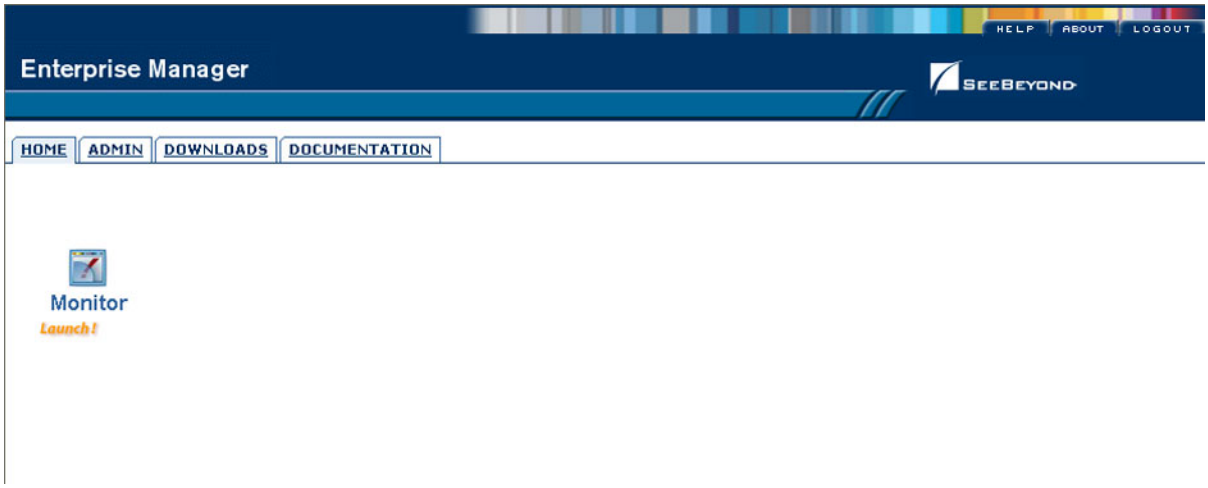
Table 3 Enterprise Manager - Tabs

Tab	Function
Help	The Help tab provides access to the online help system.
About	The About tab displays the installed version of the product. This tab is displayed only on the Home page.
Home	The Home tab returns you to the Home page. This tab is displayed on all pages other than the Home page.
Logout	The Logout tab logs you out of the Enterprise Manager and returns you to the Login page.

3.3.1 Home

The Enterprise Manager's **Home** tab (see Figure 10) contains a link to the Enterprise Monitor. Click the **Monitor** icon to launch the Monitor (see **The Enterprise Monitor** on page 31).

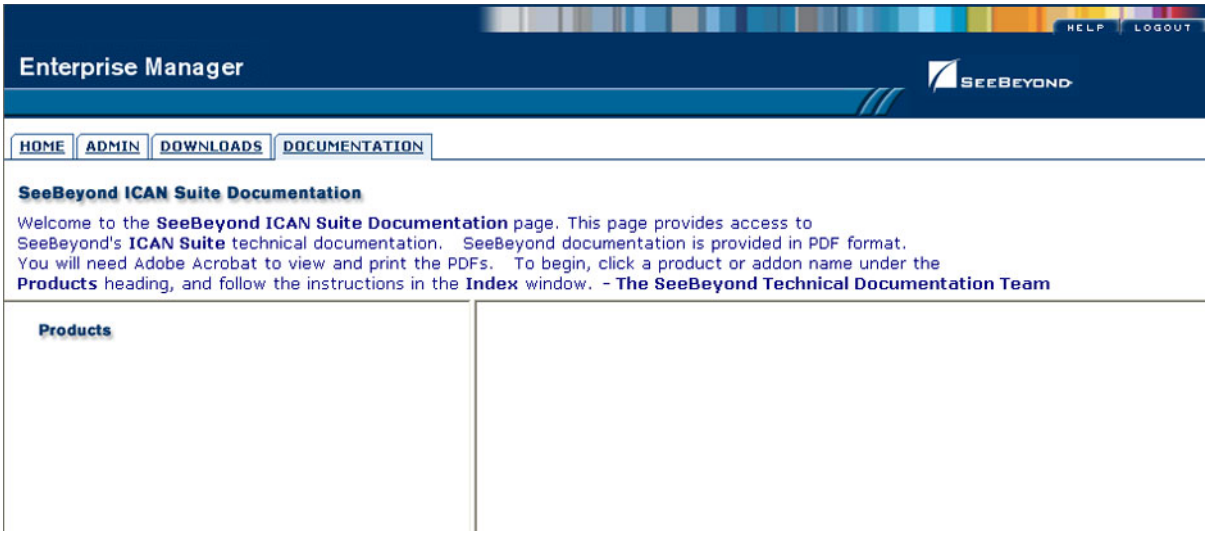
Figure 10 Monitor Launch Window



3.3.2 Documentation

The **Documentation** tab (see Figure 11) contains links to the latest versions of the SeeBeyond ICAN documentation in PDF format. You must have Adobe Acrobat Reader installed on your computer to view or print these documents.

Figure 11 Documentation Tab

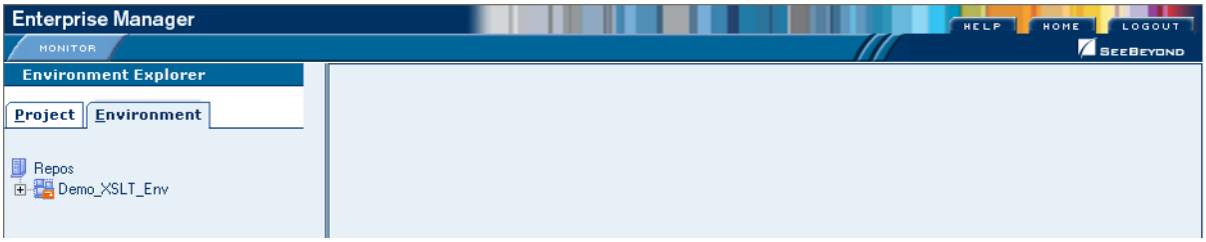


Note: *You must download the documentation SAR files from the installation disk before you can access any documents from this page (see the ICAN Installation Guide).*

3.4 The Enterprise Monitor

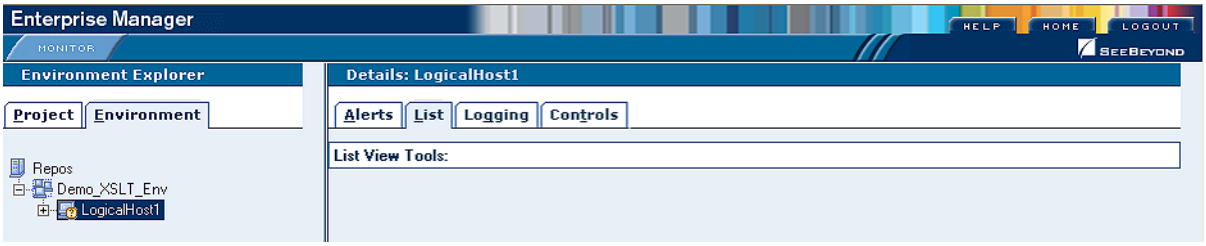
The Monitor has structure similar to that of the Enterprise Designer, with an Explorer panel on the left and a Details panel on the right. Initially, the Details panel is blank as shown in Figure 12.

Figure 12 Monitor Interface - Initial



Like the Enterprise Manager itself, the Monitor's **Details** area is organized into sections represented by tabs. Which tabs are present depends upon the component selected in the Explorer. For example, selecting the Logical Host displays the Monitor page shown in Figure 13 .

Figure 13 Monitor Interface - Environment



At times, the Details panel will have two parts, to display an additional level of information. In this case, different tabs will be displayed in the upper and lower panels. The full set of tabs is described in Table 4.

Table 4 Monitor Interface - Details Tabs

Tab	Function
Alerts	Displays functionality-related information about the component selected in the Explorer. See **Viewing Alerts** on page 33 for an example.
List	Displays a list presenting information about the component selected in the Explorer.
Logging	Displays all log messages for the component selected in the Explorer. See **Viewing Logs** on page 36 for an example.
Controls	Displays controls that allow an Administrator to intervene in the run-time process and perform tasks such as rolling transactions forward or backward. See **Editing XA Transactions** on page 39 for an example.

Table 4 Monitor Interface - Details Tabs

Tab	Function
Summary	Displays a summary of information regarding the component selected in the upper Details panel.
Consumption	Displays the number of messages processed by the component selected in the upper Details panel, and the number of messages still pending..

Note: *Not all listed tabs are always present, and tabs may be divided between an upper and lower Details panel.*

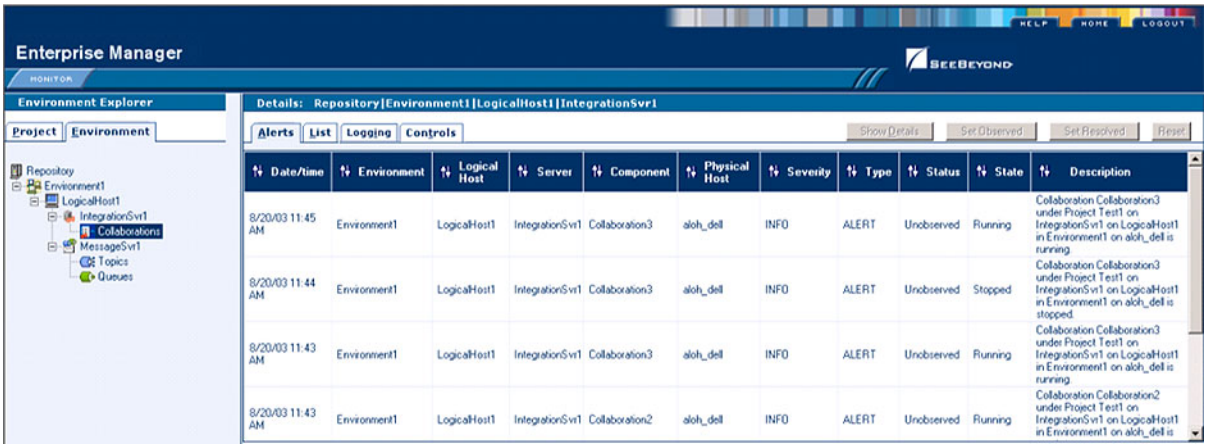
The Monitor interface offers the following viewing controls:

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

3.4.1 Viewing Alerts

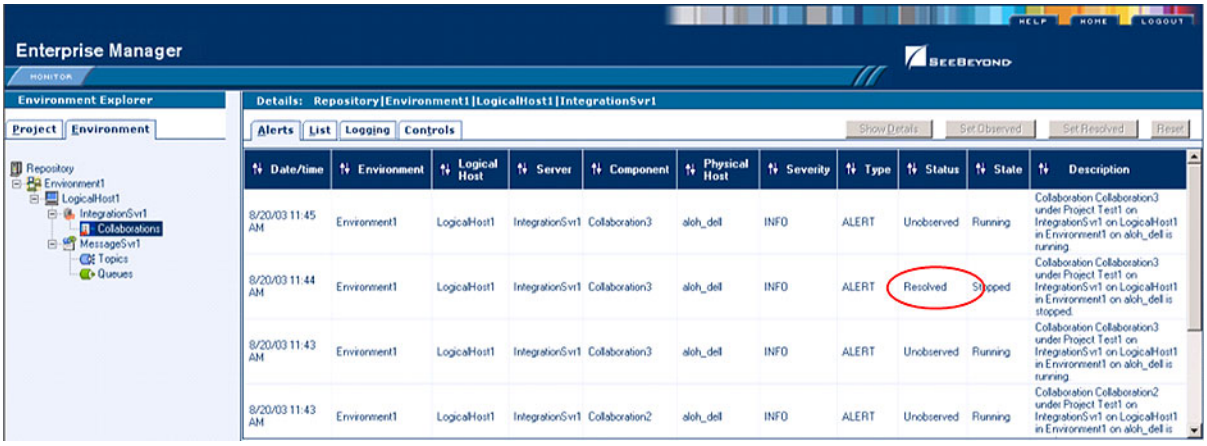
Selecting a Collaboration and clicking the **Alerts** tab in the upper Details panel displays a list of all alerts for the selected Collaboration (see Figure 14). These can be sorted by different criteria, or marked as observed or resolved.

Figure 14 Collaboration Alerts



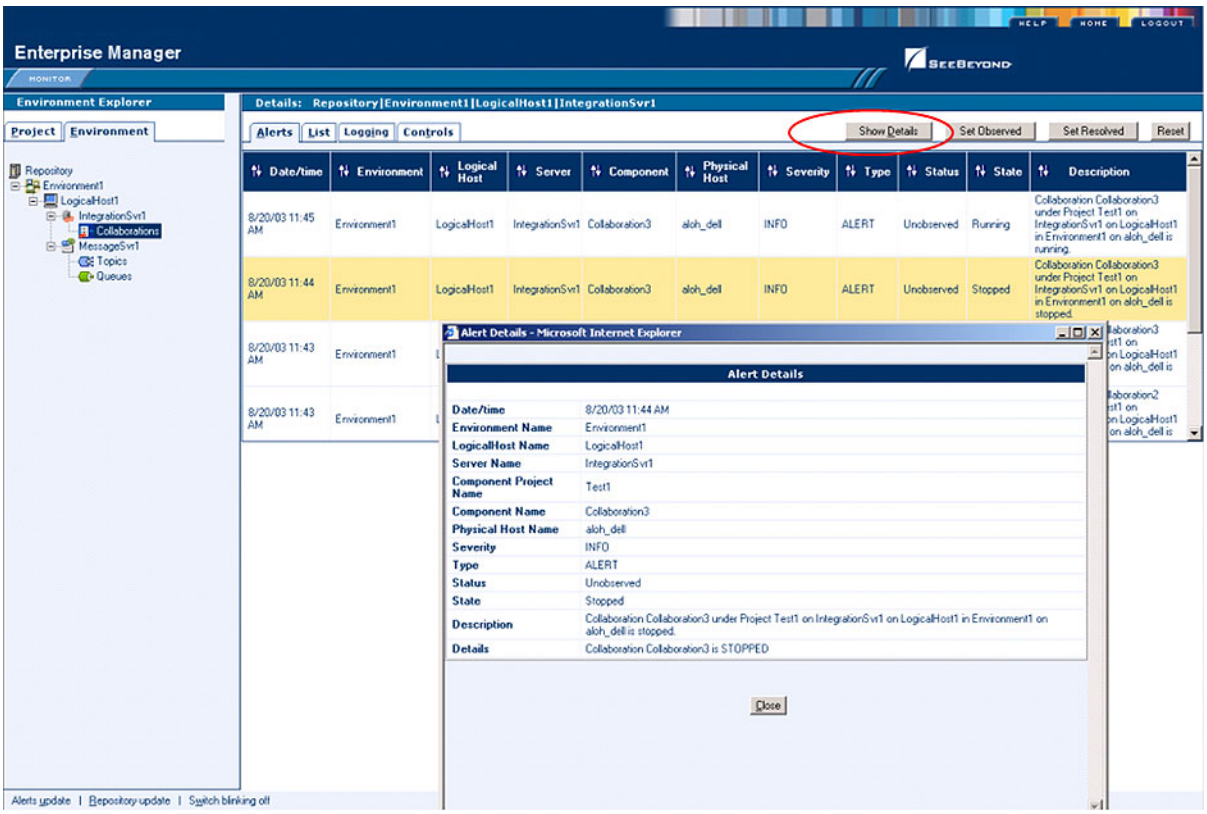
By clicking the **Set Resolved** button, you change the status of the selected Alert to *Resolved* (see Figure 15).

Figure 15 Alert Status



By clicking the **Show Details** button, you can see all details for the selected Alert (see Figure 16).

Figure 16 Alert Details



Basic default alert types are listed below. The format of the alert type is **xxx-99999**, where the first part of the code is the component designator, and the numeric part is a category code.

Table 5 Basic Alert Types

Type	Description
LH-00001	Logical Host {2} in {1} on {0} exited.
LH-00002	Logical Host {2} in {1} on {0} is already running.
LH-00003	Logical Host {2} in {1} on {0} started.
LH-00004	Logical Host {2} in {1} on {0} stopped.
LH-00005	Logical Host {2} in {1} on {0} killed.
LH-00006	Logical Host {2} in {1} on {0} is now running.
IS-00001	Integration Server {3} on {2} in {1} on {0} has exited.
IS-00002	Integration Server {3} on {2} in {1} on {0} is already running.
IS-00003	Integration Server {3} on {2} in {1} on {0} has stopped.
IS-00004	Integration Server {3} on {2} in {1} on {0} is not running (possibly crashed).

Table 5 Basic Alert Types

Type	Description
SNMP-00001	SNMP Agent has been configured.
SNMP-00002	SNMP Agent has not been configured.
COL-00001	Collaboration {5} under Project {4} on {3} on {2} in {1} on {0} is running.
COL-00002	Collaboration {5} under Project {4} on {3} on {2} in {1} on {0} is stopped.

3.4.2 Viewing Logs

Integration Server Level

In the Environment view, select an Integration Server and click the **Logging** tab in the Details panel to display all log messages for the selected component (see Figure 17). You can also filter the list for a specific log level (see Figure 18).

Figure 17 Integration Server Log Messages

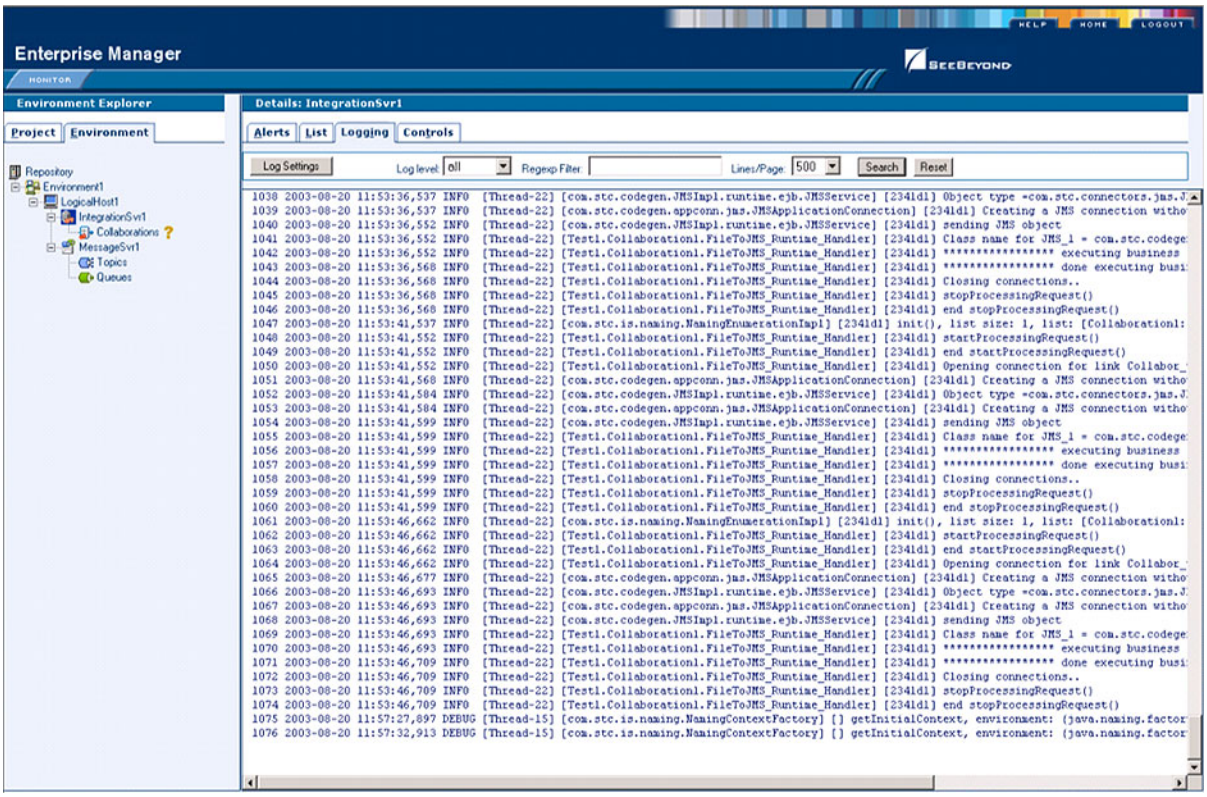
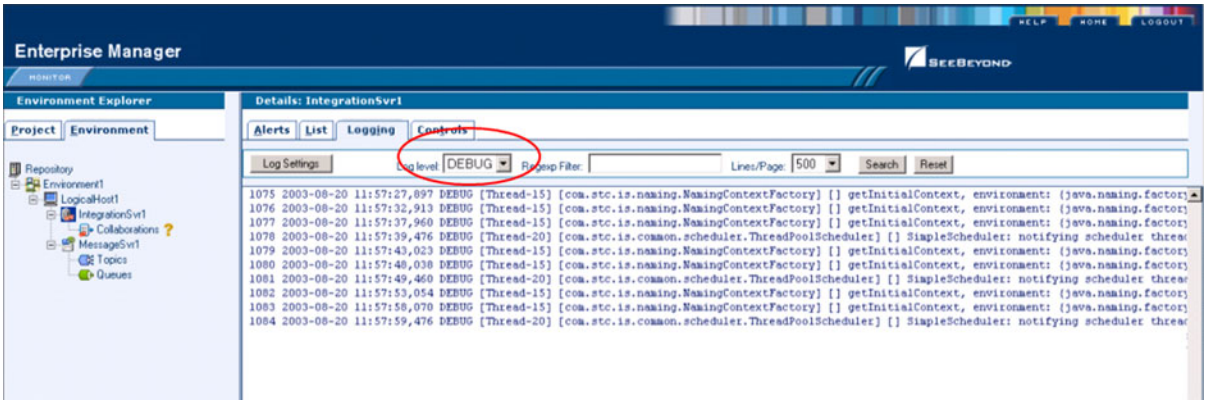


Figure 18 Integration Server Log Messages - Filtered



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 19). 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 20).

Figure 19 Collaboration Log File

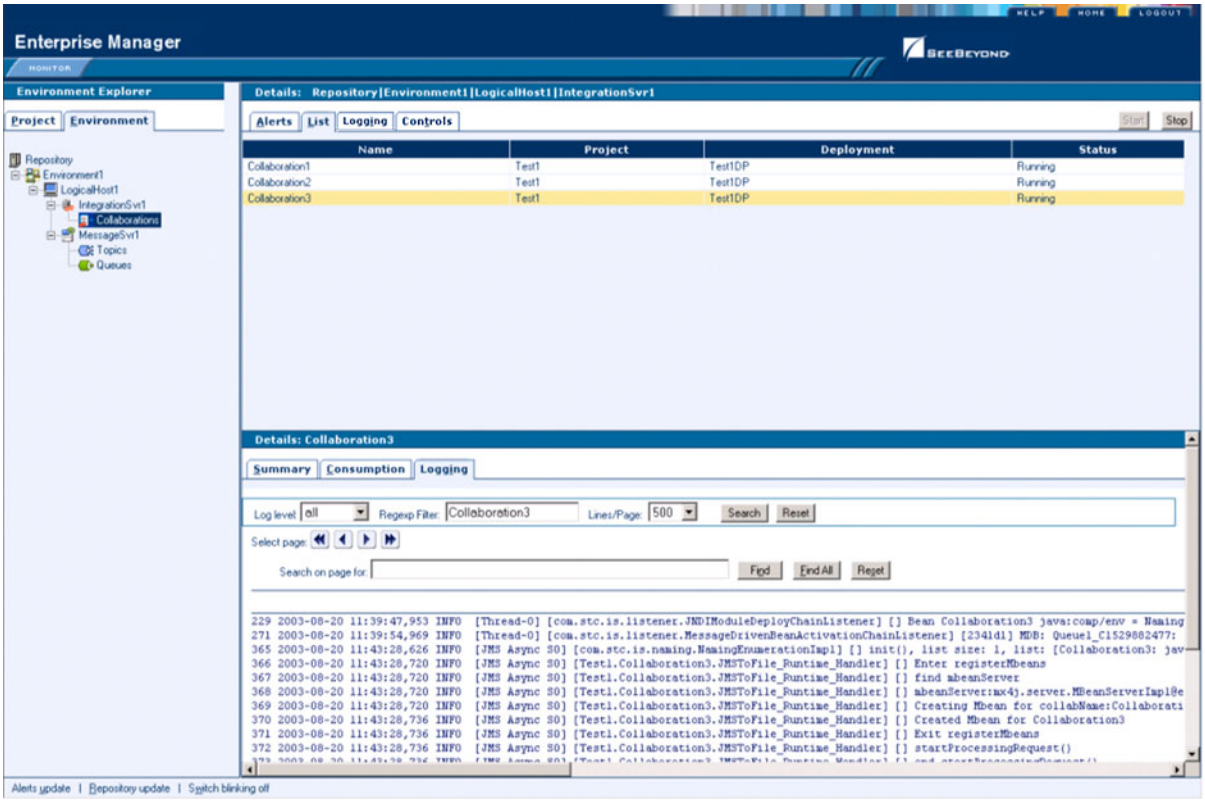
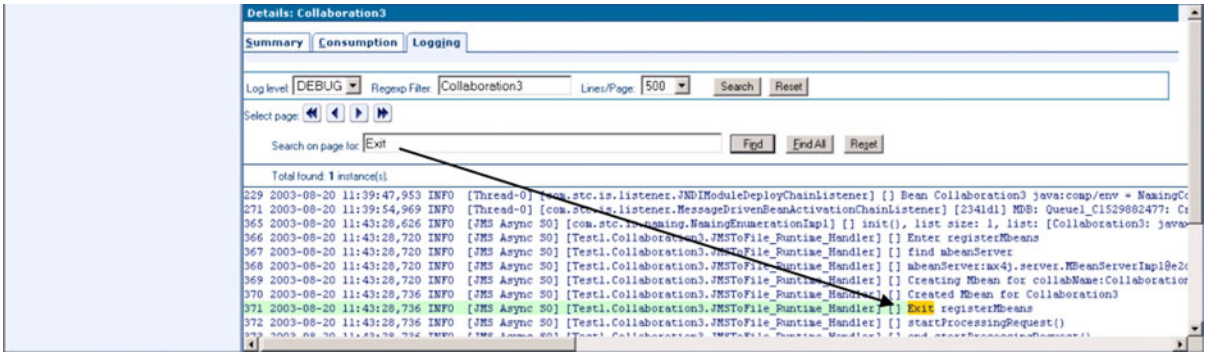


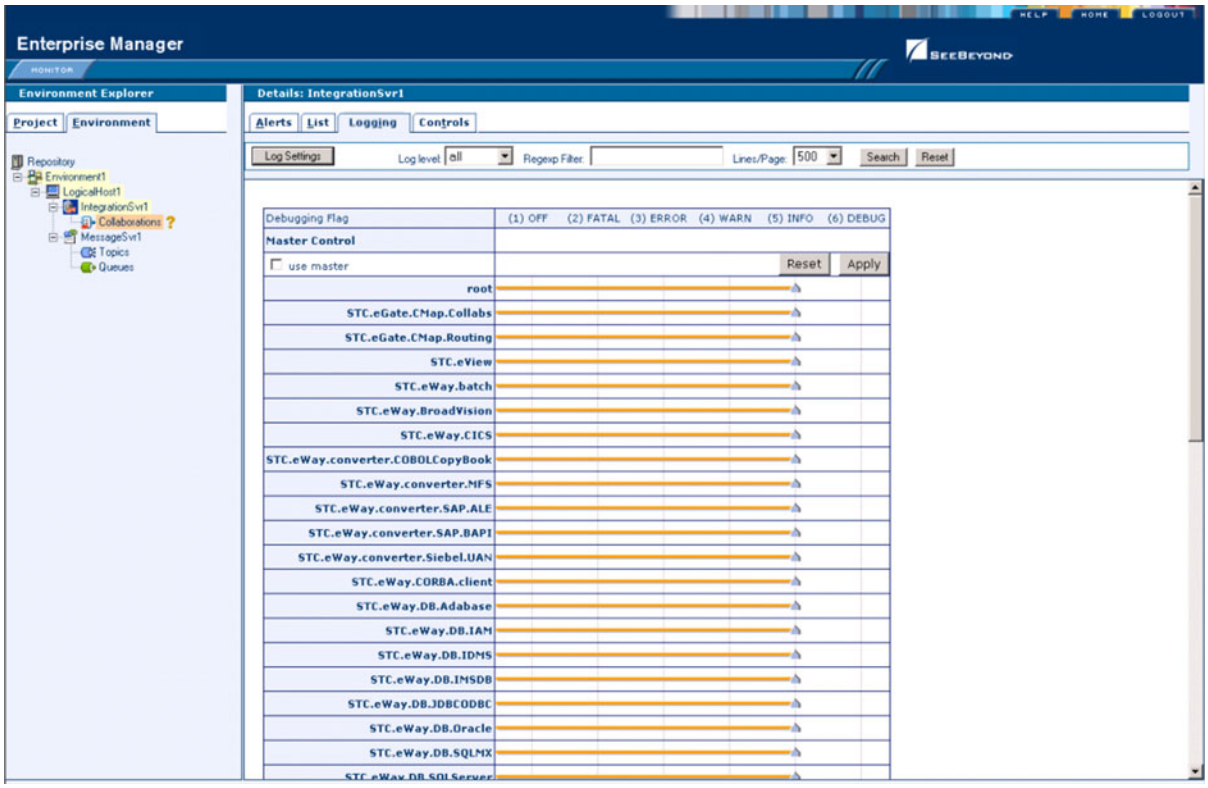
Figure 20 Search on Keyword



3.4.3 Setting Log Levels

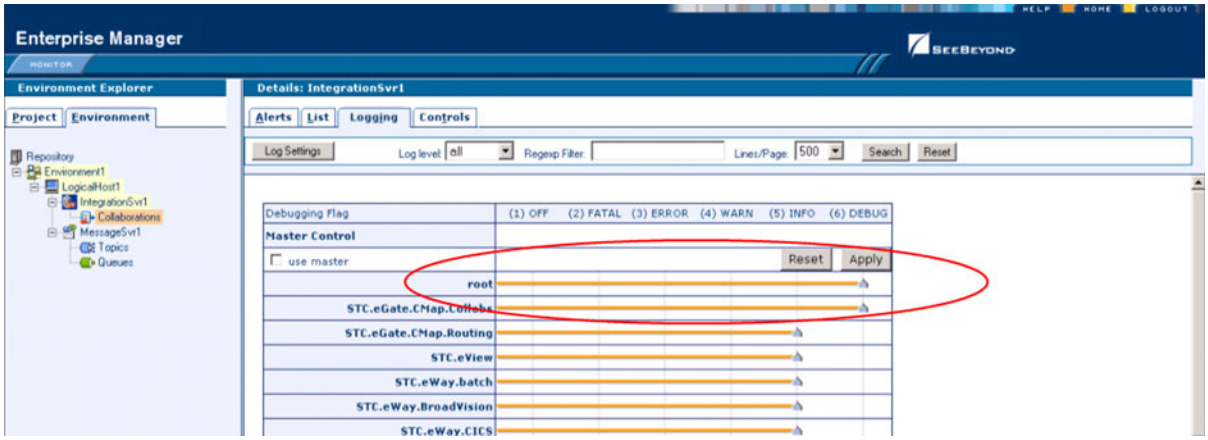
Select an Integration Server or Logical Host and click the **Log Settings** button to display the Logging Properties page (see Figure 21).

Figure 21 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 22 shows the first two properties reset to the **debug** level.

Figure 22 Resetting Log Levels



3.4.4 Editing XA Transactions

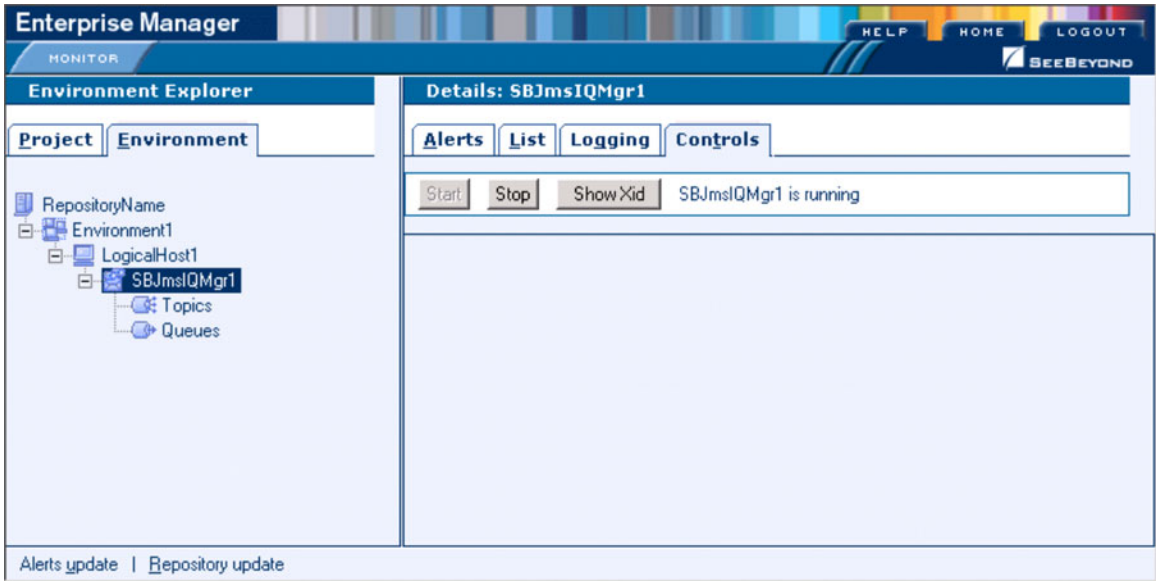
Occasionally, one of the Resource Managers (such as a database server or an external program) involved in an XA transaction will fail to commit. In that case, the transaction stays open until either the Resource Manager commits or rolls back, or the user intervenes.

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 23.

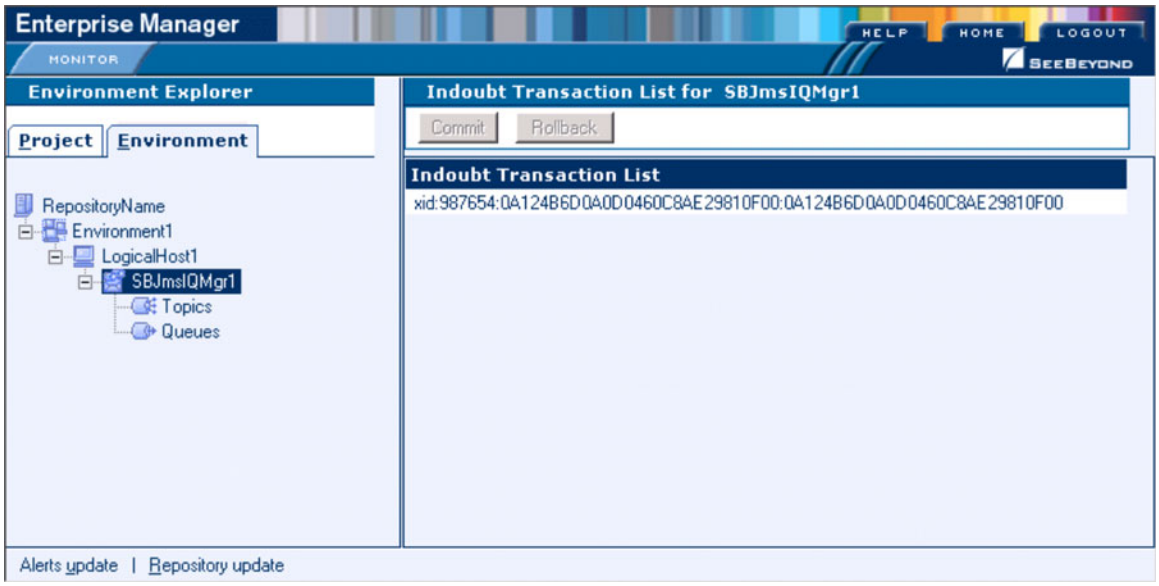
Figure 23 Message Server Details - Controls Tab



2 Click the **ShowXid** button to display the *Indoubt Transaction List* shown in Figure 24.

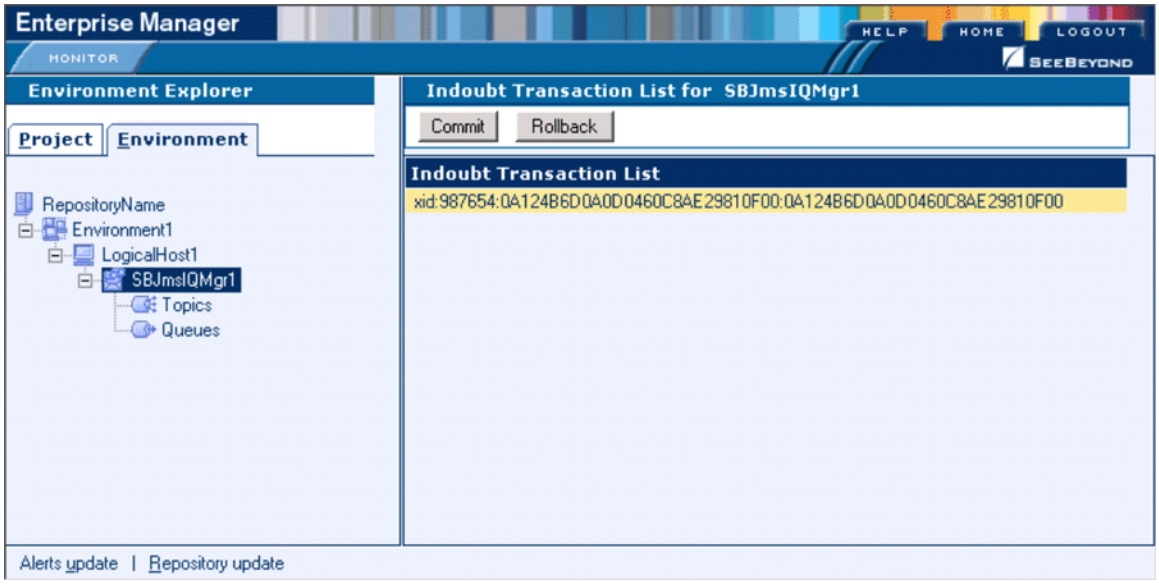
Note: *For information regarding XA transactions, see the eGate Integrator JMS Reference Guide.*

Figure 24 Indoubt Transaction List



3 Select the transaction having the specified XID, as shown in Figure 25, and click either **Commit** or **Rollback**.

Figure 25 Indoubt Transaction List - Transaction Selected



Chapter 4

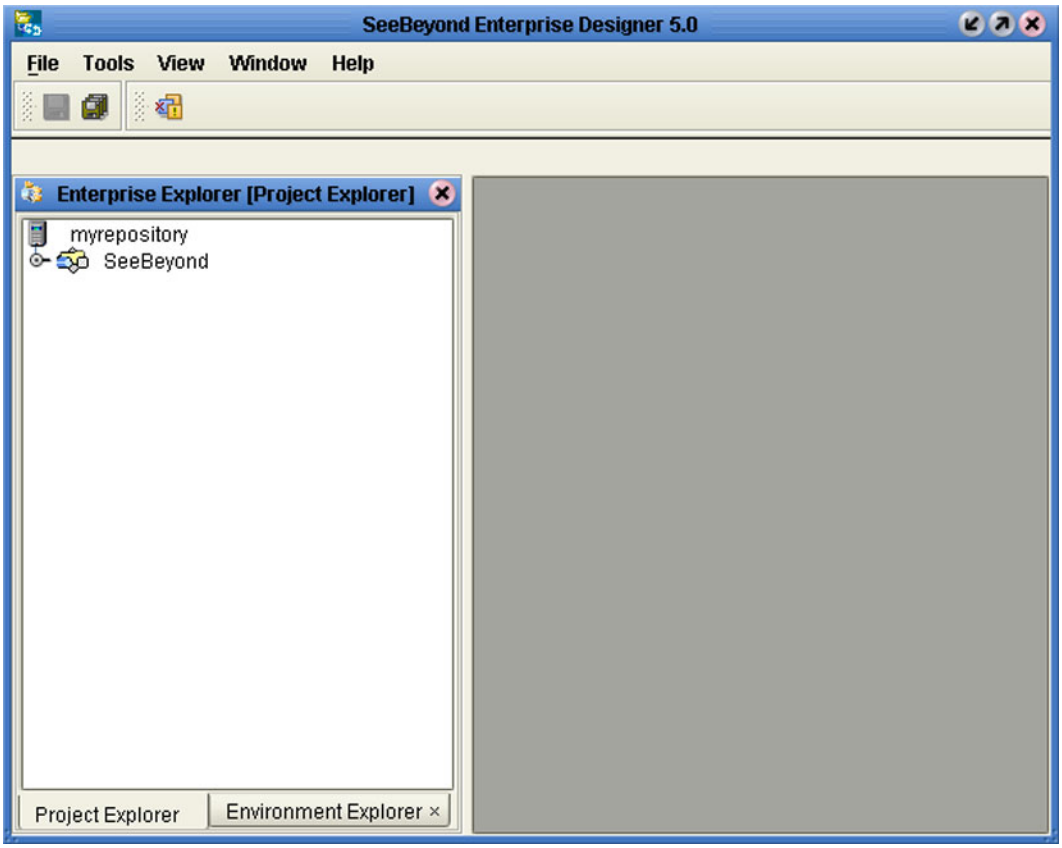
Enterprise Designer

This chapter describes the various features of the Enterprise Designer.

4.1 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 26), you can develop Projects to process and route data through an eGate Integrator system.

Figure 26 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, which are described in the following sections of this chapter:

- **Project Explorer** on page 48
- **Environment Explorer on page 49**

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 51
- **OTD Editor** on page 52
- **Environment Editor** on page 53
- **Deployment Editor** on page 54

The Enterprise Designer includes several analysis and archiving tools, which are described in the following sections of this chapter:

- **Impact Analyzer** on page 55, which helps you visualize how a change to one part of a Project would affect the rest of the Project.
- **Version Control** on page 57, which allows you to maintain multiple versions of Project components.
- **Repository Backup and Restoration** on page 60, which allows you to back up the Repository to an external file and restore a Repository from an external file.
- **Project Export and Import** on page 62, which allows you to export a Project from Enterprise Designer to an external file, and import a Project into Enterprise Designer from an external file.

The Enterprise Designer also contains the customary graphical interface features, which are described in the following sections of this chapter:

- **Menus** on page 44 describes the options contained in the individual menus.
- **Toolbar** on page 46 describes the functionality of the toolbar icons.
- **Browser Buttons** on page 47 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 43.

4.2 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,

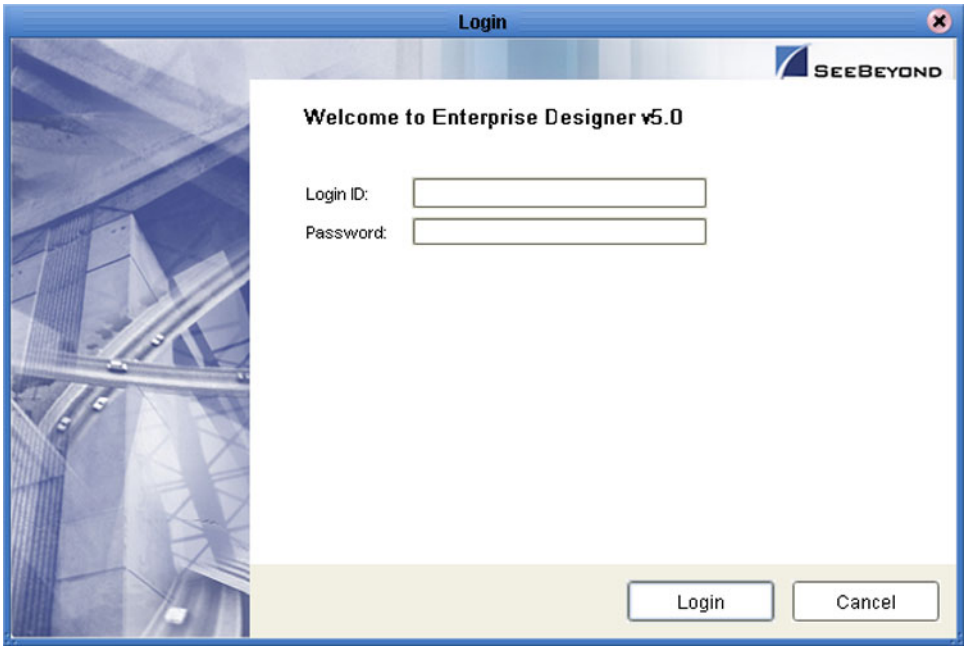
  port is the port number of the Repository, and

  rep_name is the name of the Repository.

  This displays the dialog box shown in Figure 27.

- 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 27.

Figure 27 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 26.

4.3 Interface Features

4.3.1 Menus

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

File Menu

Table 6 File Menu Options

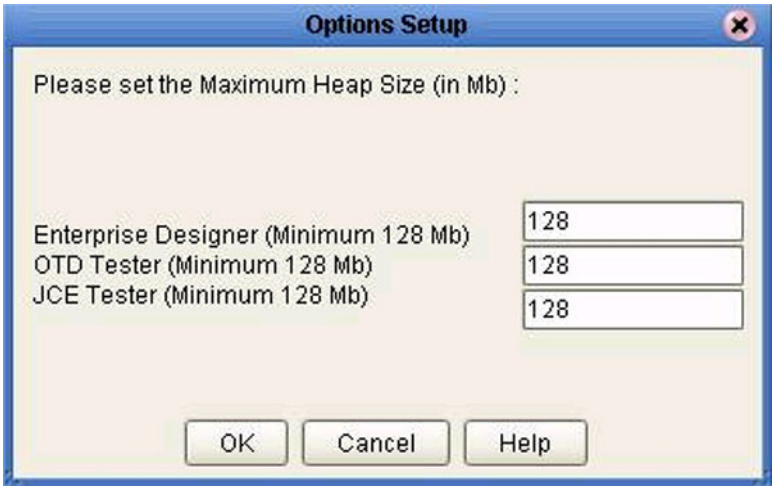
Option	Function
Restore	Displays a dialog box in which you can locate and select a Repository archive file to restore in the Enterprise Designer. See **Restoring a Repository** on page 60.
Backup	Displays a dialog box in which you can select a location on your computer to save a copy of the Repository archive. See **Backing Up a Repository** on page 60.
Save	Saves changes to the selected Project.
Save All	Saves changes to all Projects.
Exit	Closes the Enterprise Designer.

Tools Menu

Table 7 Tools Menu Options

Option	Function
Impact Analysis	Displays a dialog box in which you can view how one component of a Project impacts other components. See **Impact Analyzer** on page 55.
Update Center	Displays a series of dialog boxes in which you can check for program updates. See the *eGate Integrator Installation Guide*.

Table 7 Tools Menu Options

Option	Function
Options	Displays the Options Setup dialog box, in which you can specify the maximum heap size for selected components: 

View Menu

Table 8 View Menu Options

Option	Function
Environment Explorer	Activates the **Environment Explorer** tab on the Enterprise Explorer. See **Environment Explorer on page 49**.
Project Explorer	Activates the **Project Explorer** tab on the Enterprise Explorer. See **Project Explorer** on page 48.

Window Menu

Table 9 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.

Help Menu

Table 10 Help Menu Options

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

4.3.2 Toolbar

Table 11 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.

4.3.3 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 12 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.).

4.4 Enterprise Explorer

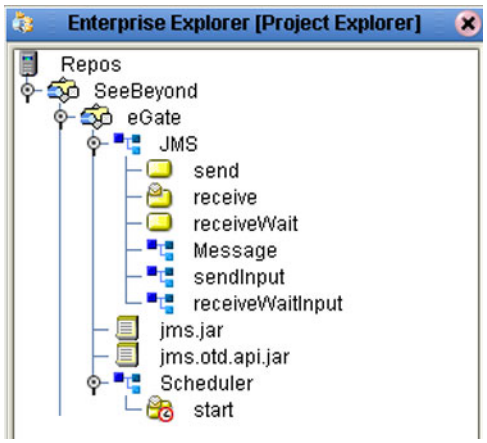
The Enterprise Explorer organizes the components of a Project into tabs that display different views of an eGate system.

- **Project Explorer** on page 48 deals with logical components.
- **Environment Explorer on page 49** deals with physical resources, including the Logical Host and Integration Server.

4.4.1 Project Explorer

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 28.

Figure 28 Enterprise Explorer: Project Explorer View



Details of the features and usage of the Project Explorer are found in **eGate Projects** on page 67.

4.4.2 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 29 Enterprise Explorer: Environment Explorer View

Details of the features and usage of the Environment Explorer are found in **Environments** on page 126.

4.5 Enterprise Designer Editors

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

- **Connectivity Map Editor** on page 51
- **OTD Editor** on page 52
- **Environment Editor** on page 53
- **Deployment Editor** on page 54

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

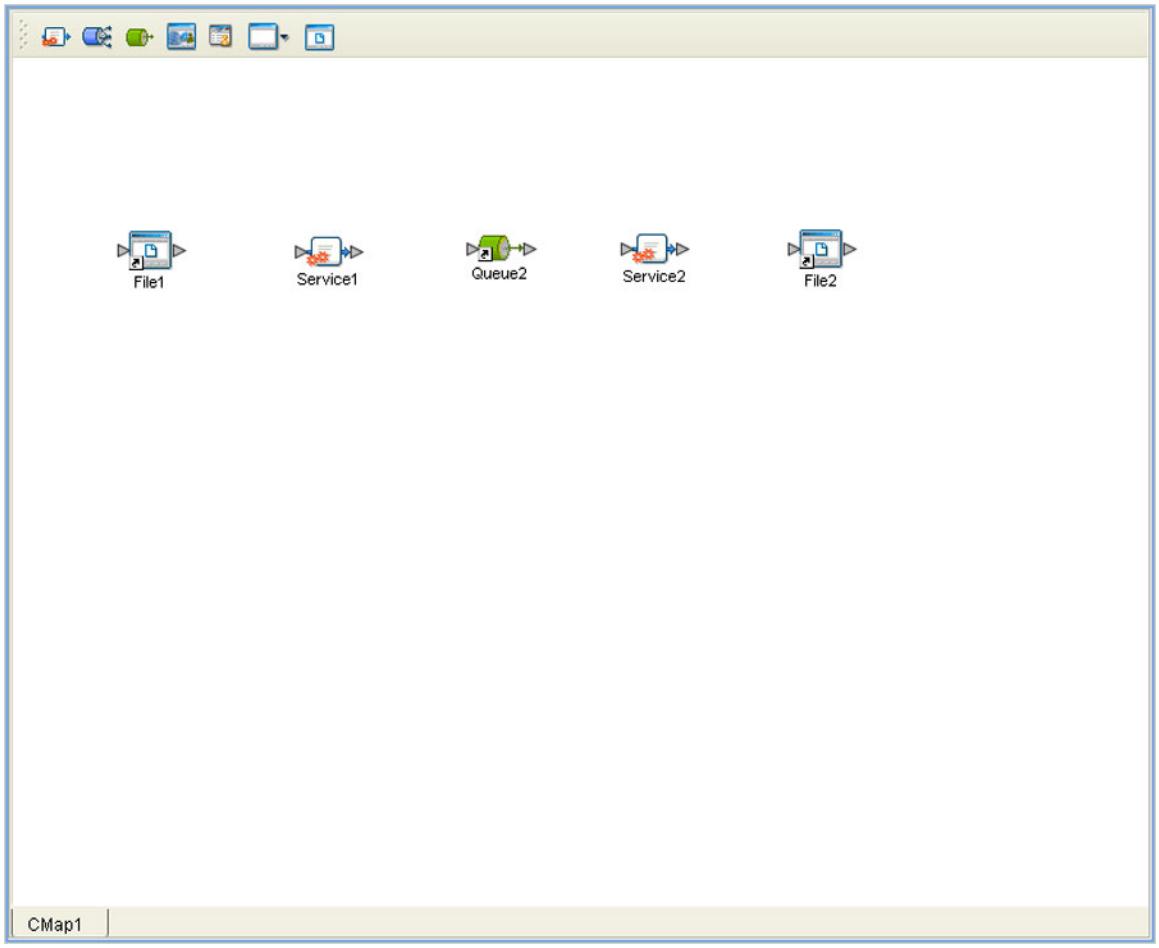
4.5.1 Connectivity Map Editor

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 30, 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 73 for detailed information.

Figure 30 Connectivity Map Editor

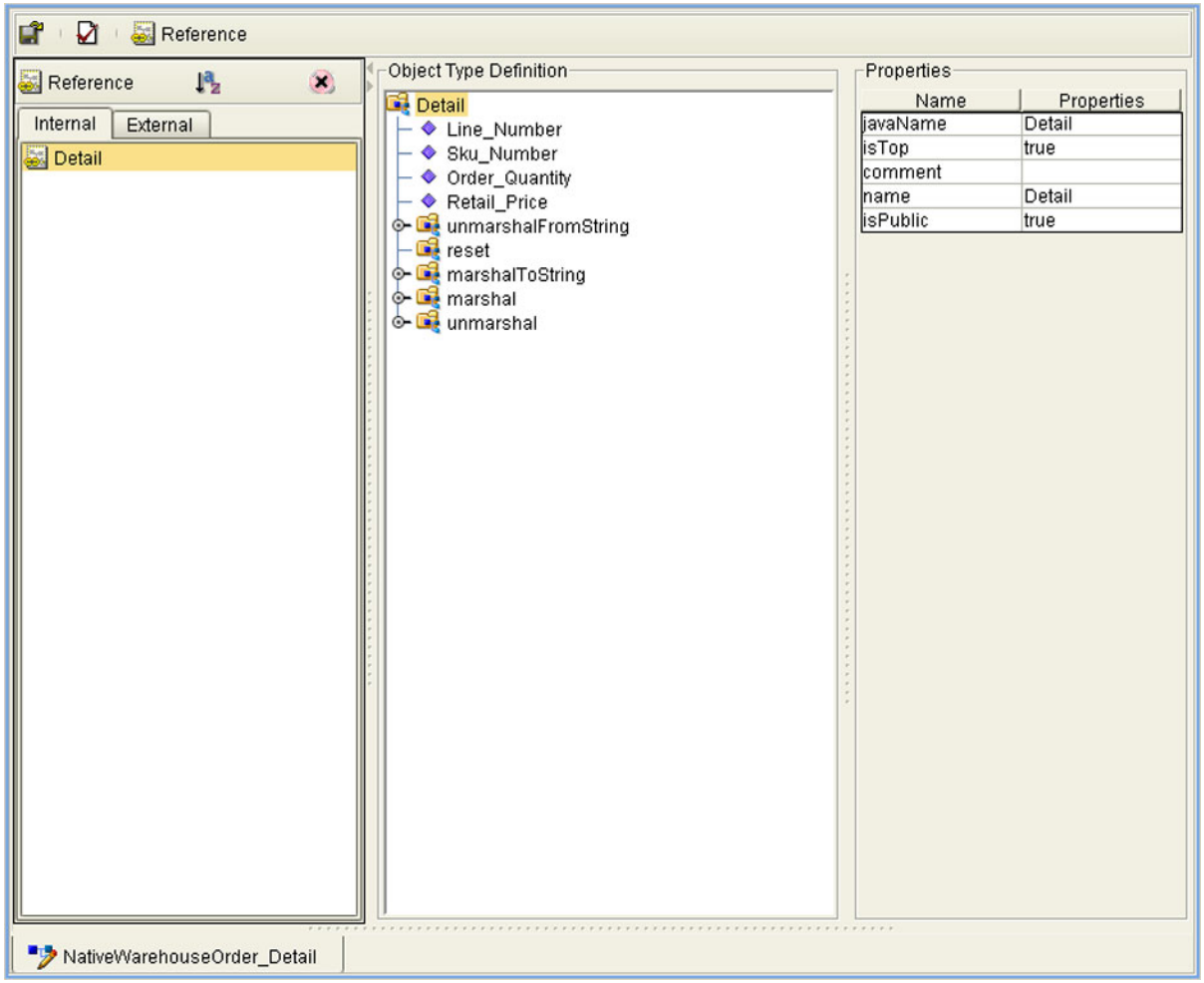


4.5.2 OTD Editor

The OTD Editor window, as shown in Figure 31, 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 120 for detailed information.

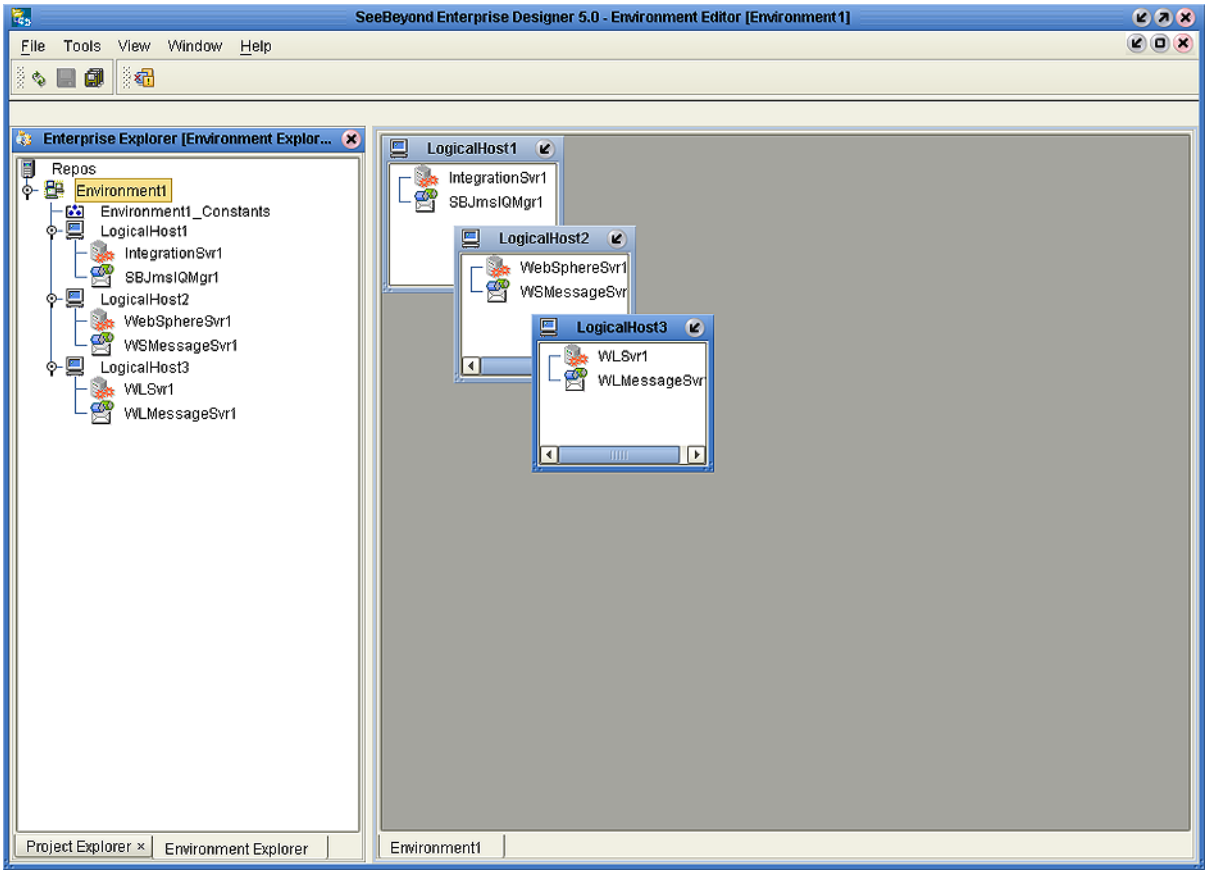
Figure 31 OTD Editor



4.5.3 Environment Editor

The Environment Editor 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 32.

Figure 32 Environment Editor

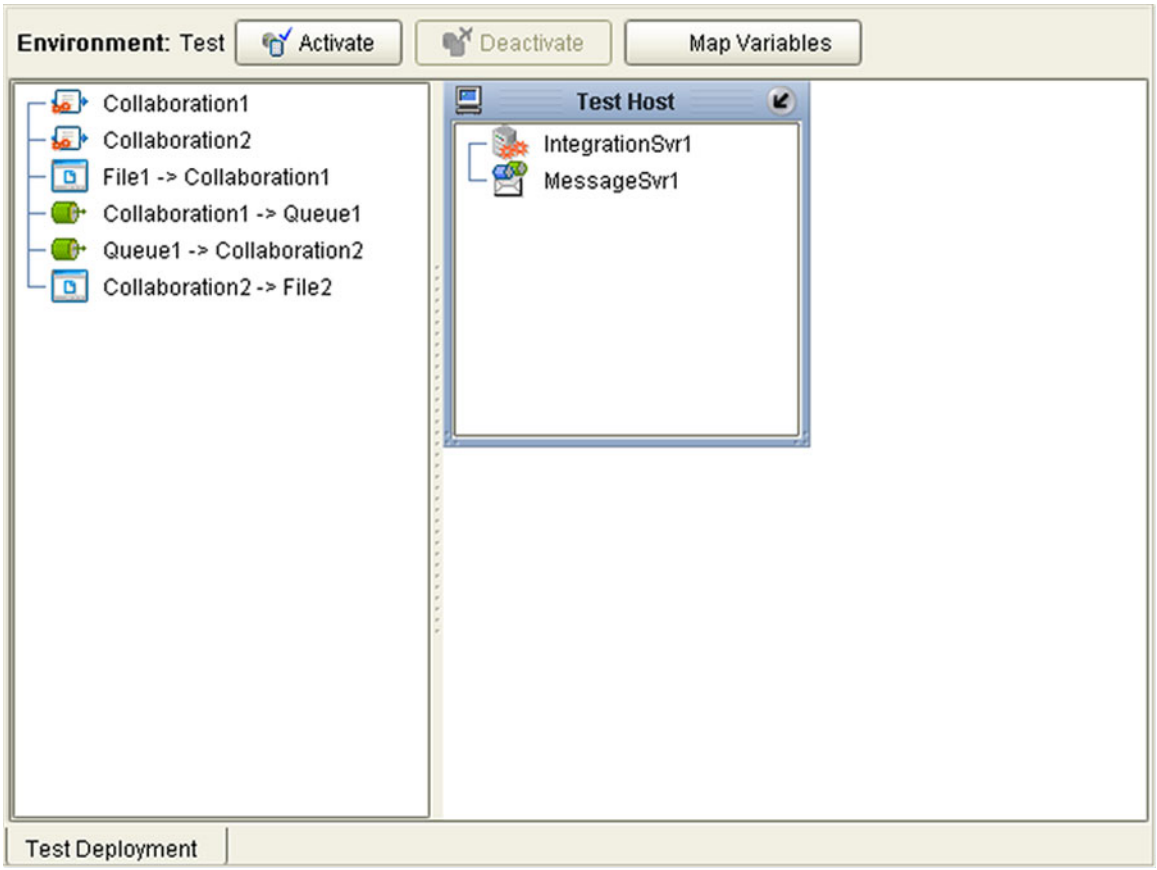


Note: *Unlike changes to Project-related configuration properties, changes to Environment-related properties do not require redeployment.*

4.5.4 Deployment Editor

The Deployment Editor, as shown in Figure 33, contains information about how Project components will be deployed in an Environment. See **Using the Deployment Editor** on page 148 for detailed information

Figure 33 Deployment Editor



4.6 Additional Tools and Features

4.6.1 Impact Analyzer

The Impact Analyzer helps you determine how a change to one component of a Project or Environment will affect other components in that Project or Environment.

To perform an Impact Analysis

1. Select a component in either the Project Explorer or Environment Explorer.
2. Click the **Impact Analyzer** button, or select **Impact Analyzer** from the Tools menu, to display the *Impact Analyzer* dialog box shown in Figure 34.
3. In the *Please show me* drop-down list, select items you would like to view.
4. From the list of objects that appears, select one for which you would like to perform an impact analysis.
   - You can print the object list by clicking **Print** to display the Windows *Print* dialog box.
5. Click **Impact** to see how that object would be affected by a change to the component you selected in step 1.

Figure 34 Impact Analyzer Dialog Box

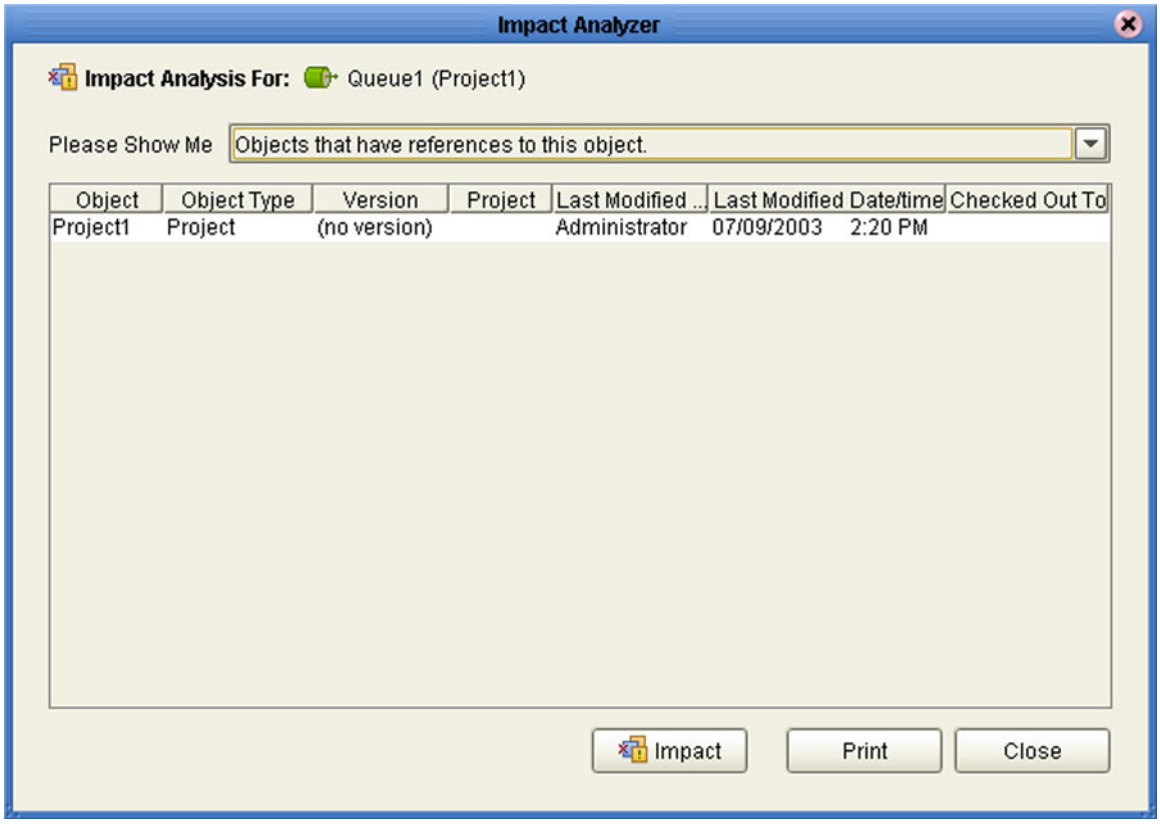
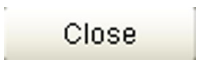


Table 13 Impact Analyzer Command Buttons

Button	Function
	Performs an impact analysis for the object selected from the object list.
	Displays the Windows **Print** dialog box, which you can use to print the object list.
	Closes the **Impact Analyzer** dialog box.

4.6.2 Version Control

Version control allows you to maintain multiple versions of a Project or Environment component. The version history of each component is recorded to a log file, and can be viewed by means of a menu option.

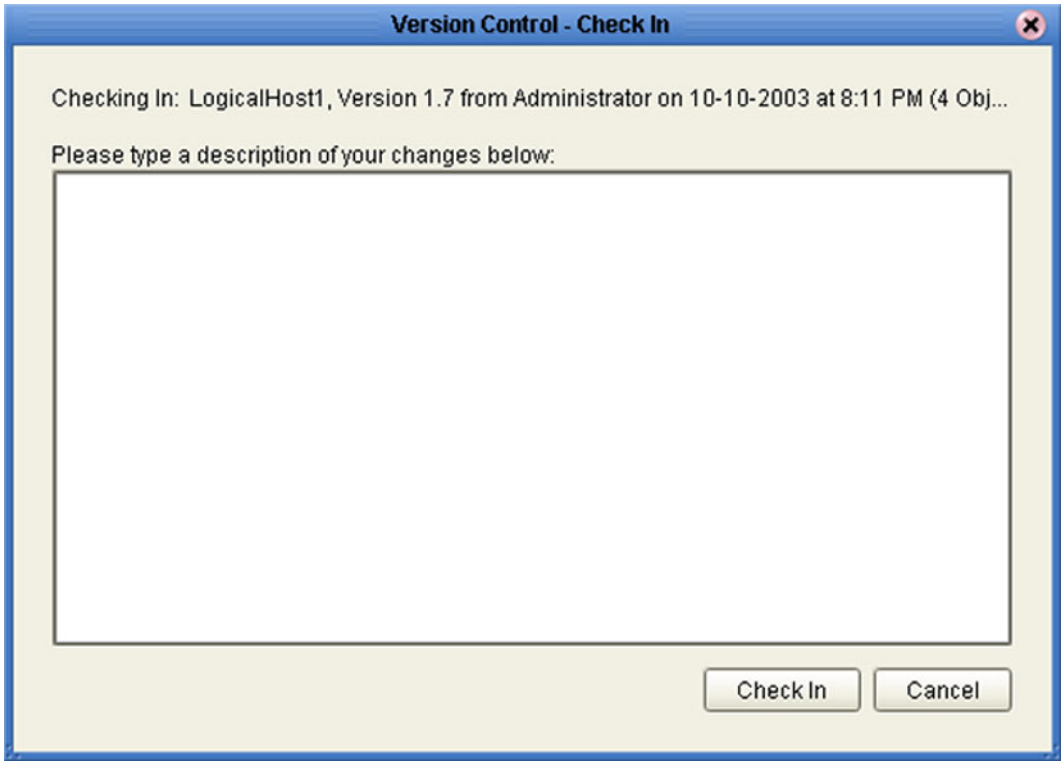
Checking a Component In

Once you have created and configured a Project or Environment component, you can check that object in by using the following procedure.

To check in a version of a Project/Environment component

1. Click the Project or Environment Explorer tab in the Enterprise Explorer.
2. Right-click on a component to display to display its context menu.
3. Select **Check In** to display the *Version Control - Check In* dialog box shown in Figure 35.

Figure 35 Version Control - Check In Dialog Box



4. Type in a description of the changes in the new version.
5. Click **Check In** to save your changes to a new version.

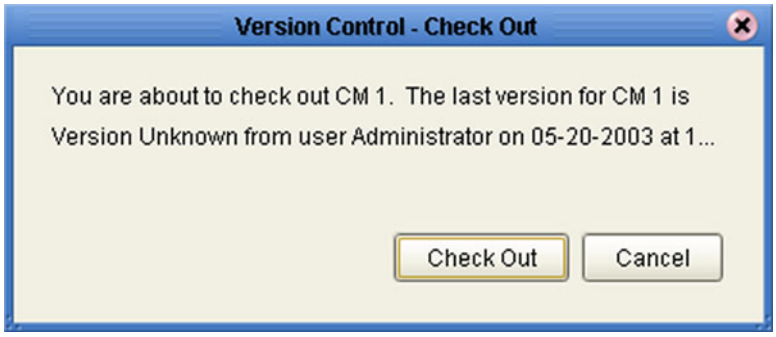
Checking a Component Out

Once an object has been checked in, you can check it out by using the following procedure.

To check out a version of a Project/Environment component

1. Click the Project or Environment Explorer tab from the Enterprise Explorer.
2. Right-click on a component to display to display its context menu.
3. Select **Check Out** to display the *Version Control - Check Out* dialog box shown in Figure 36.

Figure 36 Version Control - Check Out Dialog Box



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

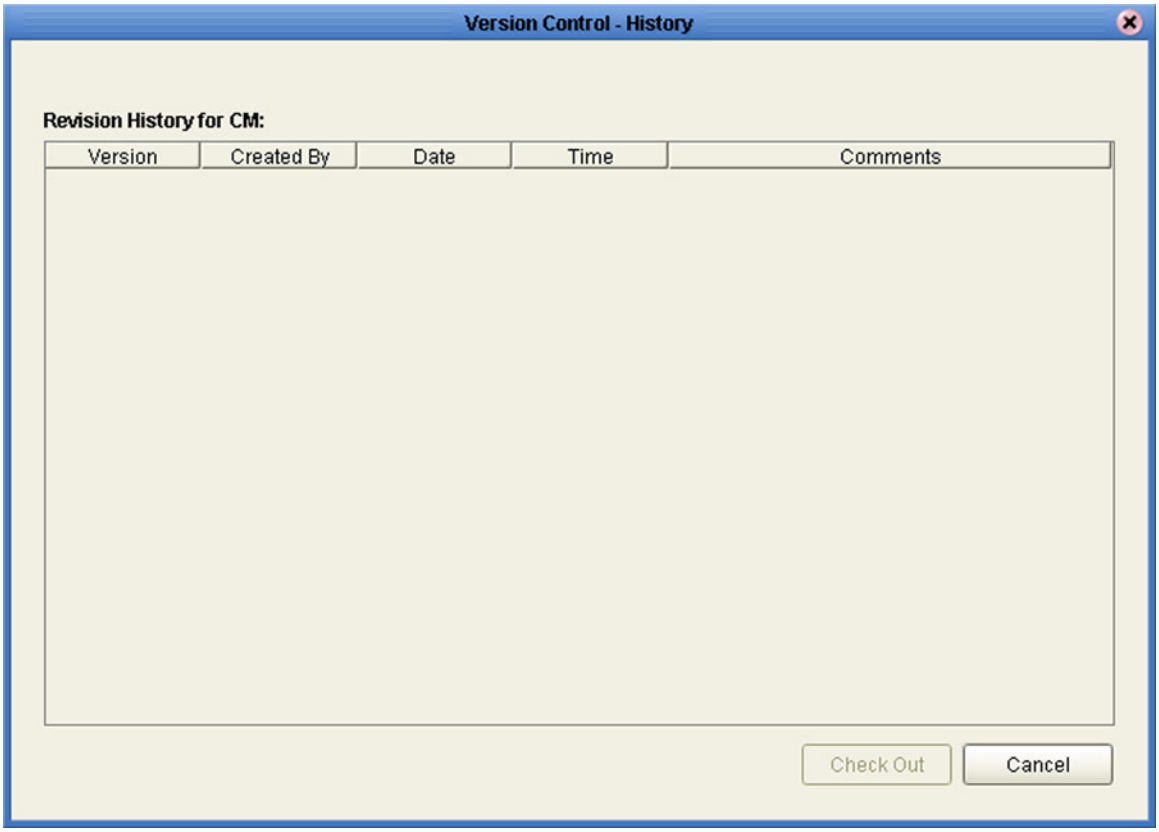
Note: *Only one user can have a file checked out for editing at a time. If another user attempts to check out the same file, they will receive a message indicating that the file is currently checked out.*

Viewing a Component's Version History

To view the version history for a component

1. Click the Project or Environment Explorer tab in the Enterprise Explorer.
2. Right-click on a component to display to display its context menu.
3. Select **Version History** to display the *Version Control - History* dialog box shown in Figure 37.

Figure 37 Version Control - History Dialog Box



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

4.6.3 Repository Backup and Restoration

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, .nbm, 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 backup.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*\util.
2. Type (for example): **backup Administrator stc c:\mybackup.zip**.

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 backup.sh)
```

Command Syntax:

```
restore username password filename
```

When restoring a Repository, note that:

- Restoring overwrites the contents of the target Repository.
- The restored Repository will have the same name as the Repository it replaced.
- 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*\util.
2. Type (for example): **restore Administrator stc c:\mybackup.zip**.
3. Restart the Repository.
4. Stop Enterprise Designer.
5. Restart Enterprise Designer.

4.6.4 Project Export and Import

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.

Note: *Logs of these processes can be found in:*

```
ICAN-root\repository\logs\repository.log
```

Exporting a Project Using Enterprise Designer

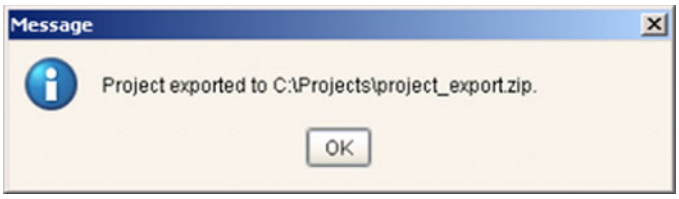
To export a Project using Enterprise Designer

1. From the Project context menu, select **Export Project ...** to display the *Enter file name to export to* dialog box shown in Figure 38.

Figure 38 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 39 is displayed when the Project file has been created. (This process takes a few minutes.)

Figure 39 Message Dialog Box



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

Exporting a Project Using the Command Line

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

Location of script file:

```
repository\util\exportProject.bat (or exportProject.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*\util.

2 Type (for example): **exportProject http://localhost:12345/myrepository Administrator stc c:\myprojectExport.zip myProject1**.

Importing a Project

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

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

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.

Note: *Logs of these processes can be found in:*

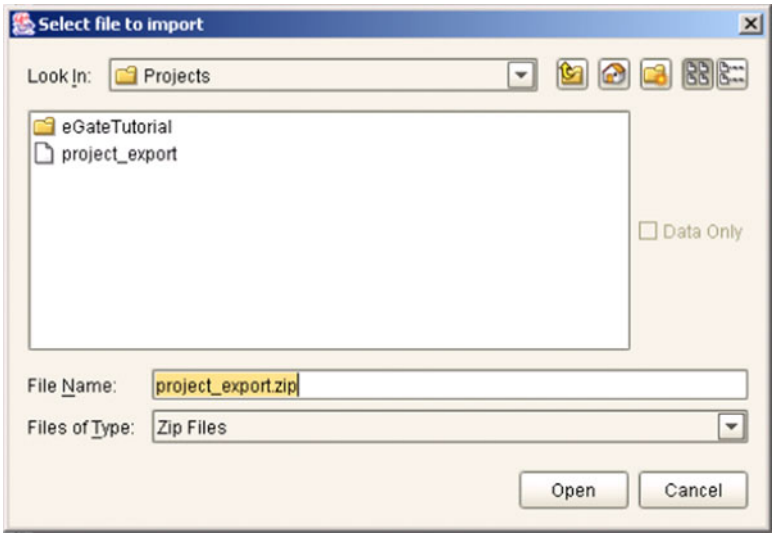
```
ICAN-root\repository\logs\repository.log
```

Importing a Project Using Enterprise Designer

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 40.

Figure 40 Select File Dialog Box

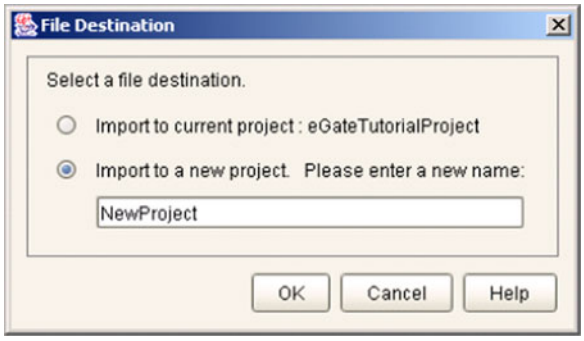


3. Locate and select the Project file that you want to import.

4 Click **Open** to import the file.

5 A File Destination dialog box will appear as shown in Figure 41, in which you choose whether to import to the same Project name or to a different name.

Figure 41 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 box shown in Figure 42 appears after the file has been imported.

Figure 42 Message Dialog Box

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

Importing a Project Using the Command Line

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

Location of script file:

```
repository\util\importProject.bat (or importProject.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*\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.

Chapter 5

eGate Projects

This chapter describes components of an eGate Project, and the use of the Enterprise Designer in defining your Project.

5.1 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, and are deployed to specific Logical Hosts in specific Environments by means of Deployment Profiles (see **Environments** on page 126). An end-to-end example of Project implementation is given in the *eGate Tutorial*.

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.

5.1.1 Project Components

The components found in a typical Project are described in the following sections of this chapter:

- **External Applications** on page 76
- **Schedulers** on page 77
- **Component Connections** on page 78
- **Message Destinations** on page 75

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, and is based on an Object Type Definition.

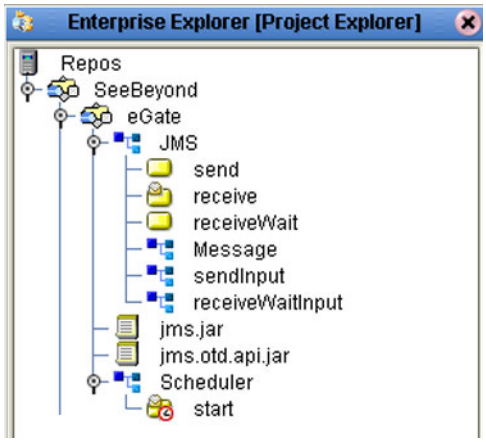
- **Object Type Definitions**

  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.

5.2 The Project Explorer

A Project consists of logical constructs and configurations designed to solve some or all of a business problem. The **Project Explorer** displays the contents of the Repository that belong to the selected Project (see Figure 43).

Figure 43 Project Explorer



The Project Explorer is used in conjunction with the Connectivity Map Editor (see **Using the Connectivity Map Editor** on page 73) to create and configure a Project.

Note: *Select **Refresh All from Repository** before you **Open** any component (such as a Collaboration) to ensure that you open the latest version of that component.*

5.2.1 Project Explorer Icons

The icons described in Table 14 appear in the Project Explorer.

Table 14 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.

5.2.2 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 70
- **Project Menu** on page 71
- **Connectivity Map Menu** on page 72

Repository Menu

Figure 44 Repository Menu

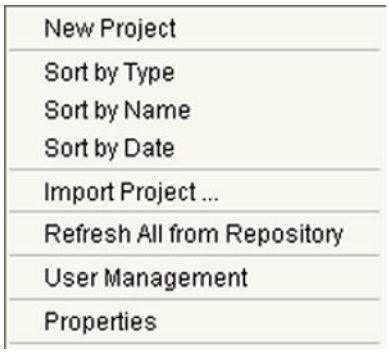


Table 15 Repository Menu Options

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 Project 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 163.
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 163.

Note: *Select **Refresh All from Repository** before you **Open** any other component (such as a Collaboration) to ensure that you open the latest version of the component.*

Project Menu

Figure 45 Project Menu

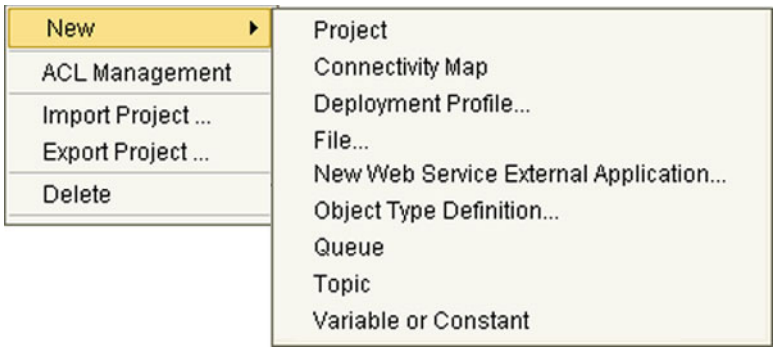


Table 16 Project Menu Options

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 73.
	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 148.
	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 **SeeBeyond Web Services** on page 86.
	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 105 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 190.
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 46 Connectivity Map Menu

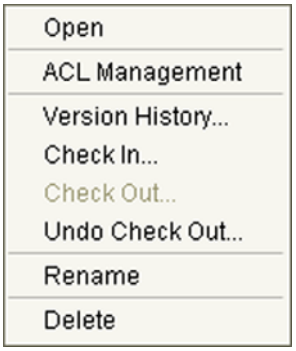


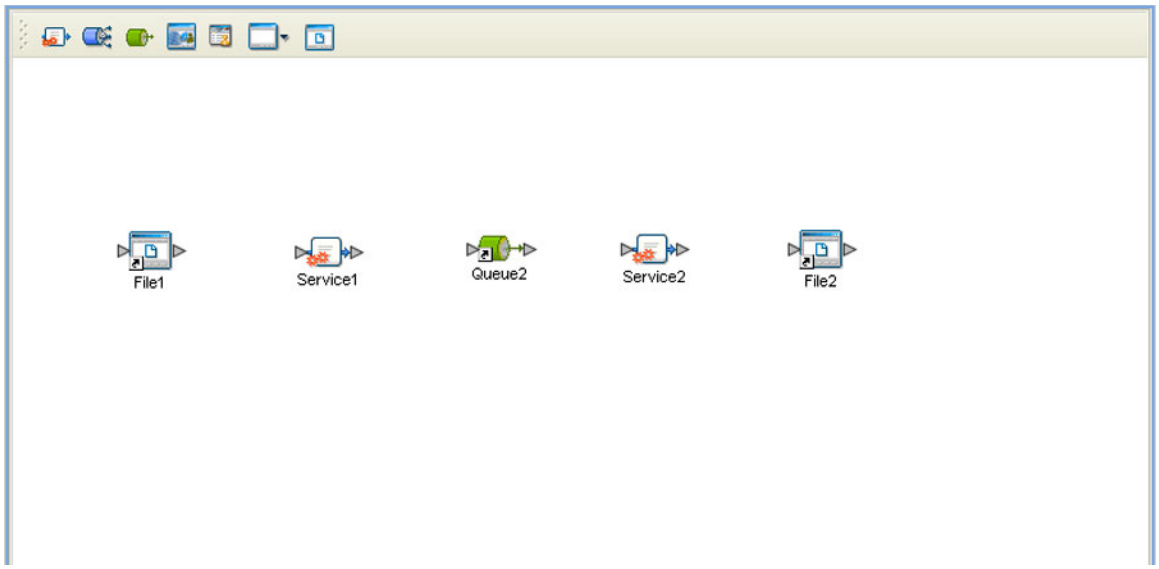
Table 17 Connectivity Map Menu Options

Command	Function
Open	Opens the **Connectivity Map Editor** for the selected Connectivity Map. See **Using the Connectivity Map Editor** on page 73.
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 190.
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 **Viewing a Component's Version History** on page 59 for more information.
Check In	Displays a dialog box, with which you can check in the current version of a Project. Refer to **Checking a Component In** on page 57 for more details.
Check Out	Displays a dialog box with which you can check out a version of a Project. See **Checking a Component Out** on page 58 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.

5.3 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 47). 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.

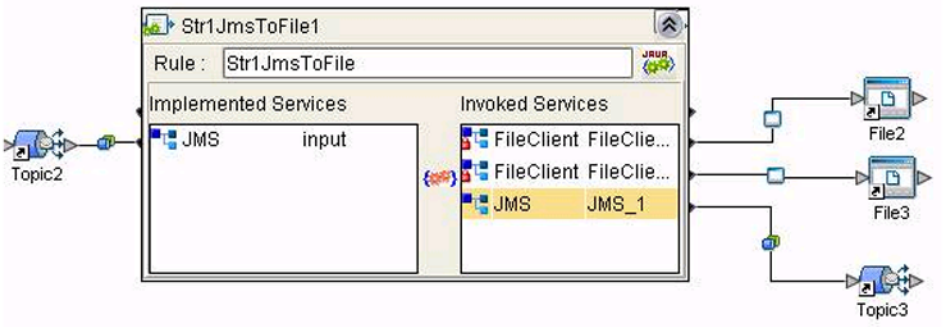
Figure 47 Connectivity Map Window



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.

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 48).

Figure 48 Linking JMS Topics



The Connectivity Map Editor toolbar contains the icons listed in Table 18, plus additional icons representing eGate add-ons and other ICAN components that you may have installed.

Table 18 Connectivity Map Toolbar Icons

Icon	Component	Function
	Service	A logical component that provides the framework for an eInsight business process.
	Queue	A Message Destination that conforms to the point-to-point messaging paradigm, having one sender and one receiver. See the *eGate Integrator JMS Reference Guide* for information.
	Topic	A Message Destination that conforms to the publish/subscribe messaging paradigm, having one sender (publisher) and multiple receivers (subscribers). See the *eGate Integrator JMS Reference Guide for* information.
	Web Service External Application	Represents a third-party Web service application external to eGate. See **SeeBeyond Web Services** on page 86.
	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 76.
	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 77.

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.

5.4 Message Destinations

A Message Destination is a container for stored data, and can follow either the topic or queue JMS model.

5.4.1 Topics

A *topic* is a message destination that conforms to the publish-and-subscribe messaging paradigm.

5.4.2 Queues

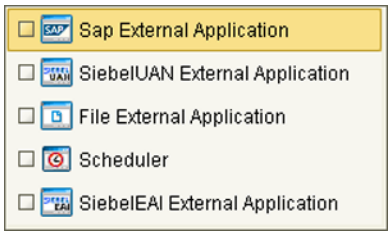
A *queue* is a message destination that conforms to the point-to-point messaging paradigm.

5.5 External Applications

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.

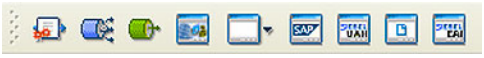
External applications are logical representations of external software applications that are being integrated by the eGate system. These are linked to a Service by means of an eWay. Clicking the drop-down arrow beside the external application icon displays a menu showing those applications corresponding to eWays that have been purchased and installed, plus the Scheduler. An example is shown in Figure 49.

Figure 49 External Application Drop-Down Menu



Checking the individual external applications adds the selected icon to the toolbar (see Figure 50).

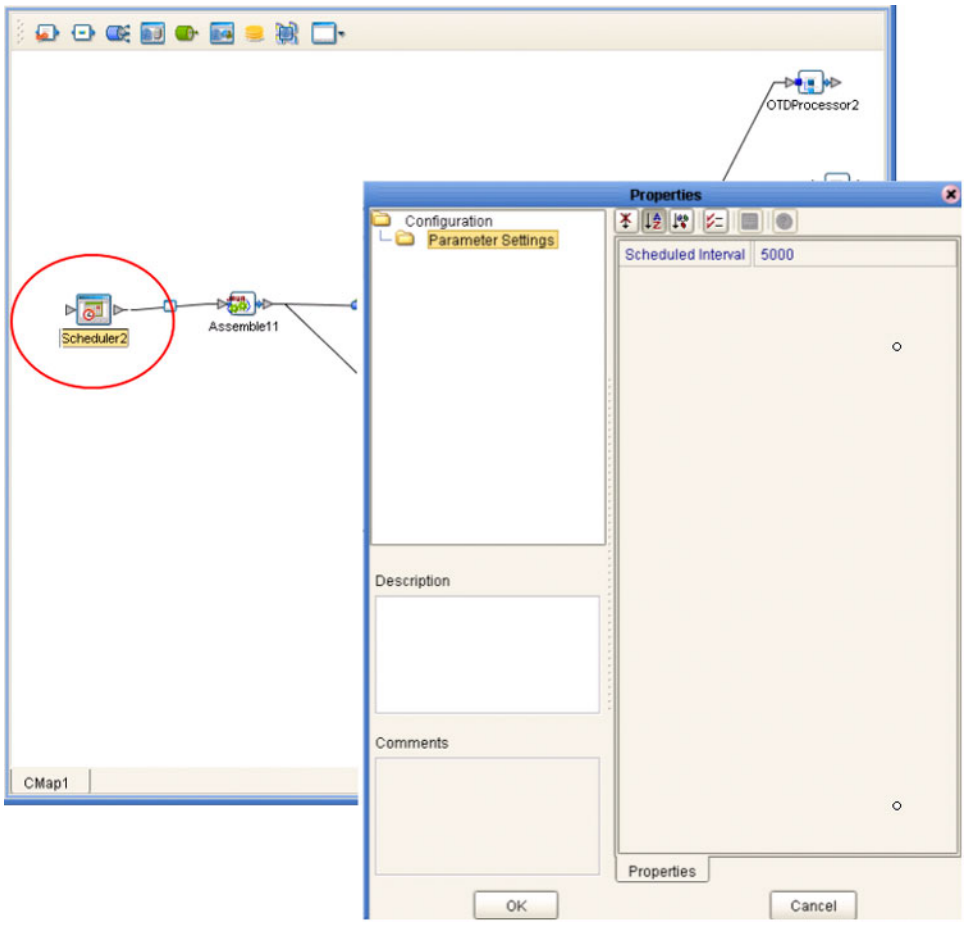
Figure 50 Selected External Applications in Toolbar



5.6 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 51 Scheduler Component

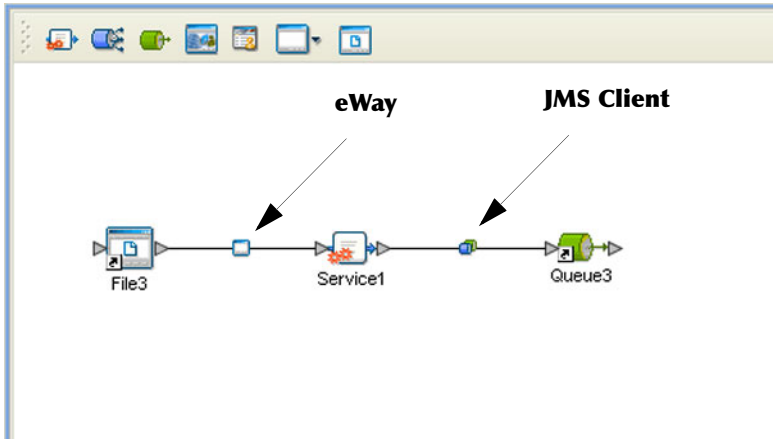


As an example, depicted in Figure 51, a scheduler can be used to invoke a Collaboration with an interval of 5 seconds (5000 milliseconds).

5.7 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 52).

Figure 52 Connection Icons in a Connectivity Map



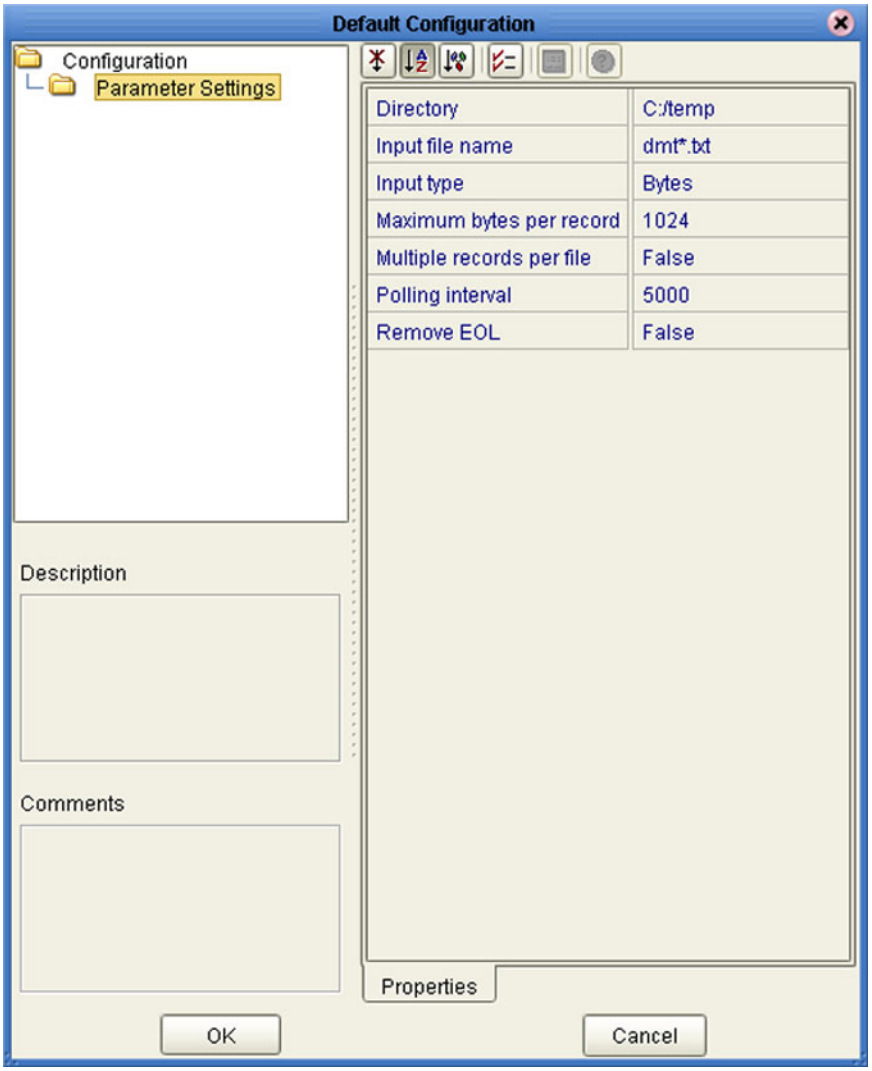
- When you link an external application with a Collaboration, the Enterprise Designer automatically adds an eWay Connection icon to the link. The eWay enables communication and movement of data between the external application and the eGate system. The eWay configuration specifies the logical connection 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 results in an error at a later time.*

5.7.1 Configuring a Connection

Double-clicking an eWay or JMS Client connection icon in the Connectivity Map displays the Default Configuration dialog box. As an example, Figure 53 shows a dialog box that lists the configuration properties for a File eWay.

Figure 53 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 **Toolbar** contains a series of buttons used to sort and modify the information listed in the Properties folder, as described in Table 19.
- 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 19 Configuration Dialog Box 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.

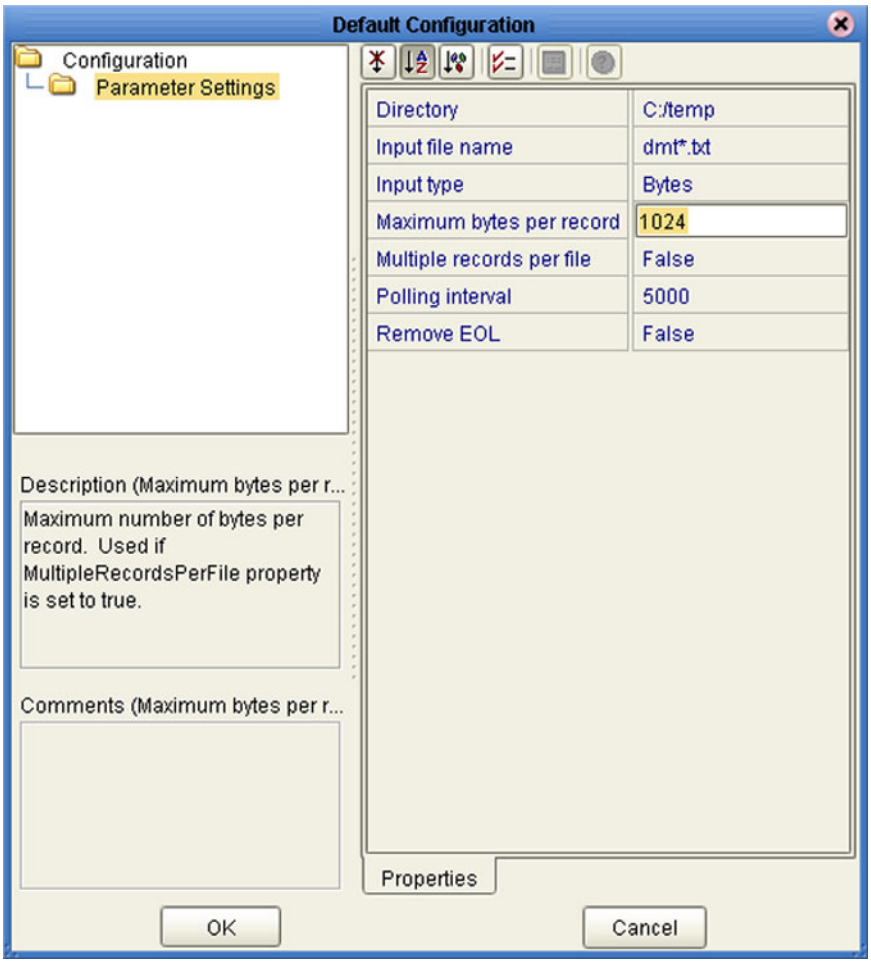
5.7.2 Modifying a Configuration Property

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 54, which highlights the current value.

Figure 54 Editing a Default Configuration Field



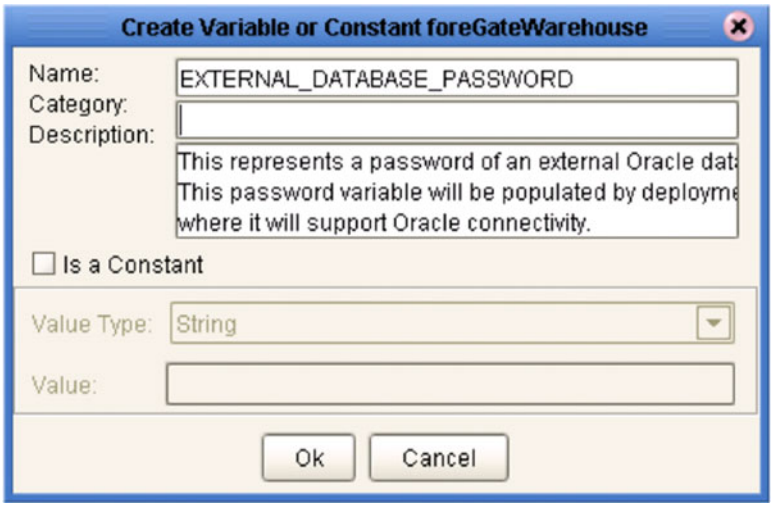
3. Enter a new value in the field. (Note that in this example, the parameter is not used unless you allow multiple records per file, as explained in the Description box).
4. Click **OK** to save the new settings.

5.8 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 154). Project variable values can be literals or Environmental constants.

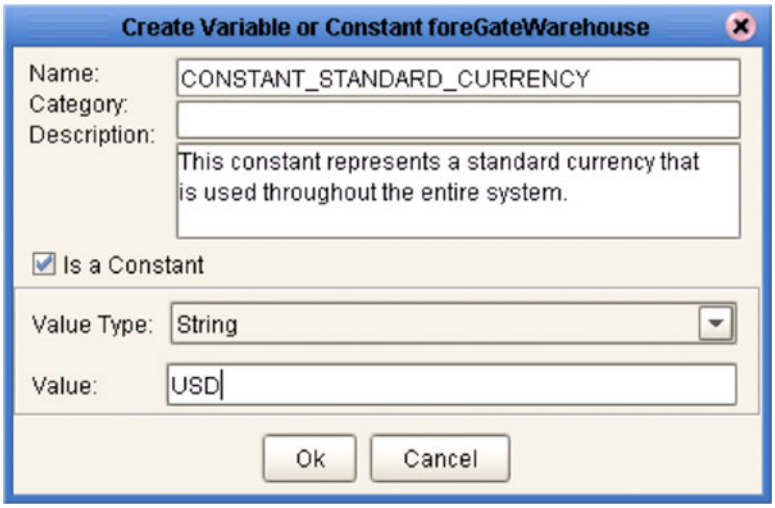
For example, a project variable is defined to represent a password of a database user in a target environment (see Figure 55). System managers will assign an actual value to this variable in the deployment profile editor. The value of the assigned project variable—an Environment constant— is then used to connect the database in the target environment.

Figure 55 Project Variable Creation



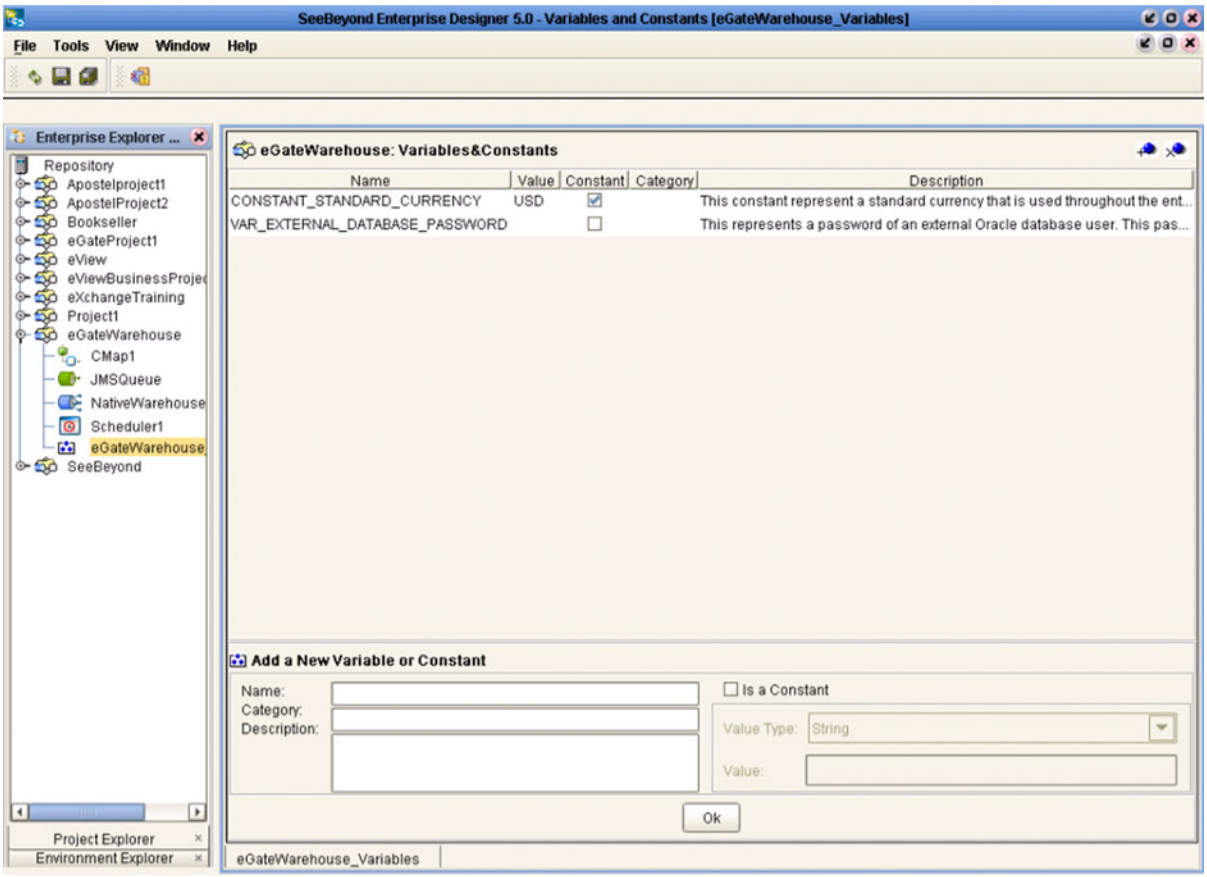
A constant is a name-value pair; when you create a constant you assign a permanent value to it, which cannot be overridden. An example of this would be a standard currency used globally throughout the system (see Figure 56).

Figure 56 Project Constant Creation



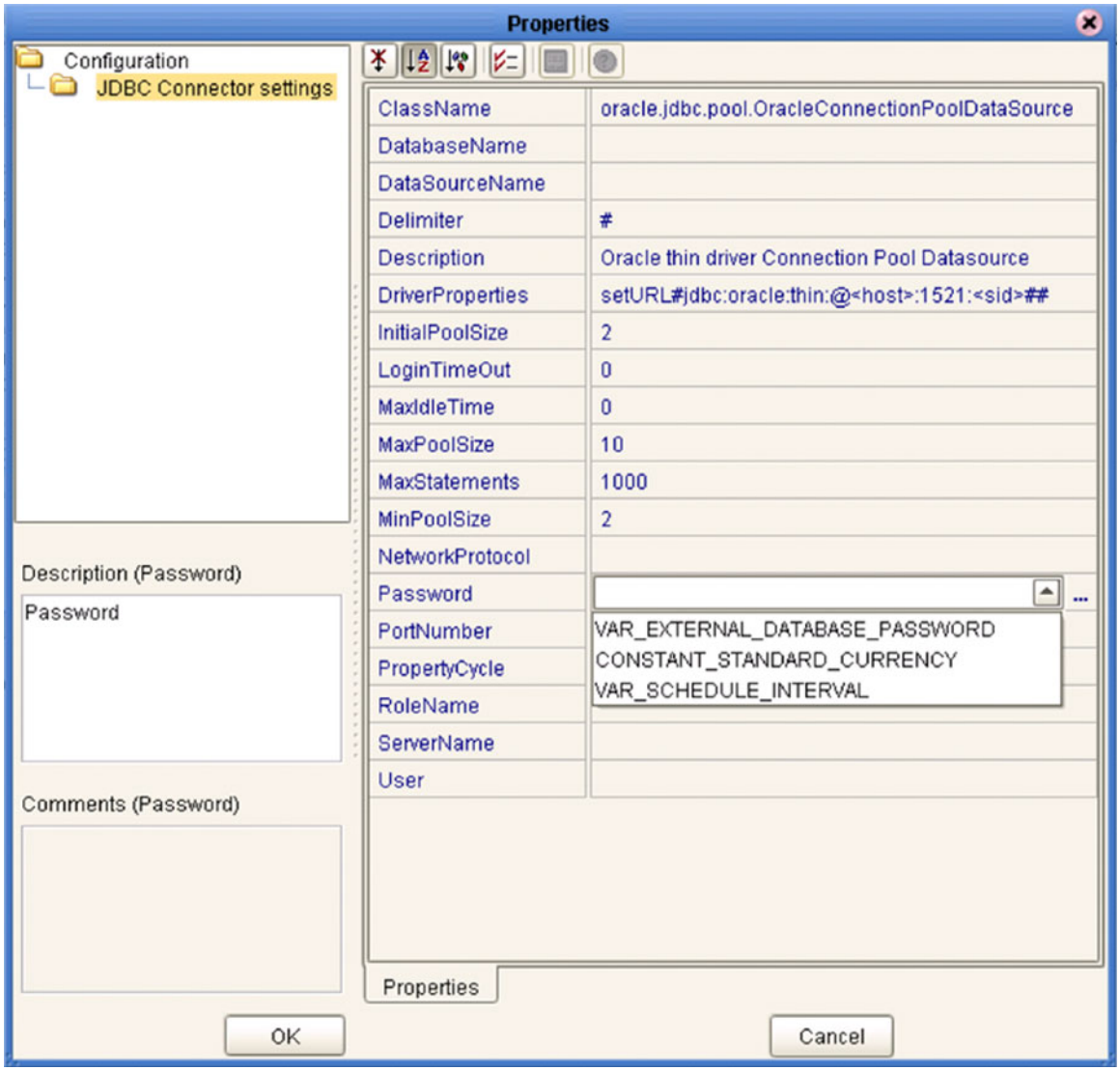
These constants and variables are automatically added to a Variables and Constants object group within the Project (see Figure 57).

Figure 57 Variables and Constants Object Group



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 58).

Figure 58 Connector Properties



See **Mapping Variables** on page 154 for information on setting the values for Project variables.

Chapter 6

Web Services

This chapter describes the use of the Web Services capability of eGate Integrator, acting with other components of the ICAN Suite.

6.1 Overview

Basically, Web Services enables communication and data transfer between diverse applications using the Internet. In doing so, it provides a means for implementing EAI (Enterprise Application Integration) within an organization, or B2B (Business-to-Business) integration between partner organizations. This capability is achieved by wrapping back-end systems to present a common, standardized interface to the connecting network.

Four related technologies are used to transform and transport data within Web Services:

- **XML** (Extensible Markup Language)

  Provides a language for defining both the data itself and the way to process it.

- **WSDL** (Web Services Description Language)

  Defines the interfaces, data types, interactions, and mappings used in the Web Services. WSDL files are used to invoke and operate Web services on the Internet and to access and invoke remote applications and databases.

- **SOAP** (Simple Object Access Protocol)

  Defines a communications envelope that is mappable to HTTP and provides a format for transmitting XML documents over a network.

- **UDDI** (Universal Description, Discovery, and Integration)

  Provides a mechanism for storing and categorizing information that allows publication of services and discovery of external services.

6.2 SeeBeyond Web Services

eGate Integrator provides the capability to create either a client or a server to receive WSDL file from a remote server, or send WSDL files to a remote client. eGate works together with eInsight Enterprise Service Bus, in which the associated business processes are developed. See **Building a Web Client** on page 89 and **Building a Web Server** on page 97.

The ICAN Suite contains the following components that implement the Web Services capability:

- **WSDL Wizard**

  The WSDL Wizard creates an OTD from a WSDL file. See **Using the WSDL Wizard** on page 111.

- **WSDL Editor**

  See the *eInsight Enterprise Service Bus User's Guide*.

- **WSDL Interface Designer**

  See the *eInsight Enterprise Service Bus User's Guide*.

- **WSDL Viewer**

  See the *eInsight Enterprise Service Bus User's Guide*.

- **UDDI Repository**

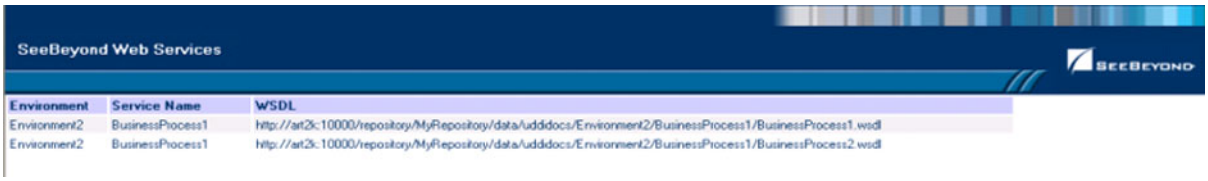
  All ICAN objects that can be accessed as Web services are presented in the SeeBeyond UDDI Repository. See **UDDI Repository** on page 87.

6.3 UDDI Repository

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 59). The URL of this repository is:

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

Figure 59 SeeBeyond UDDI Repository



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.

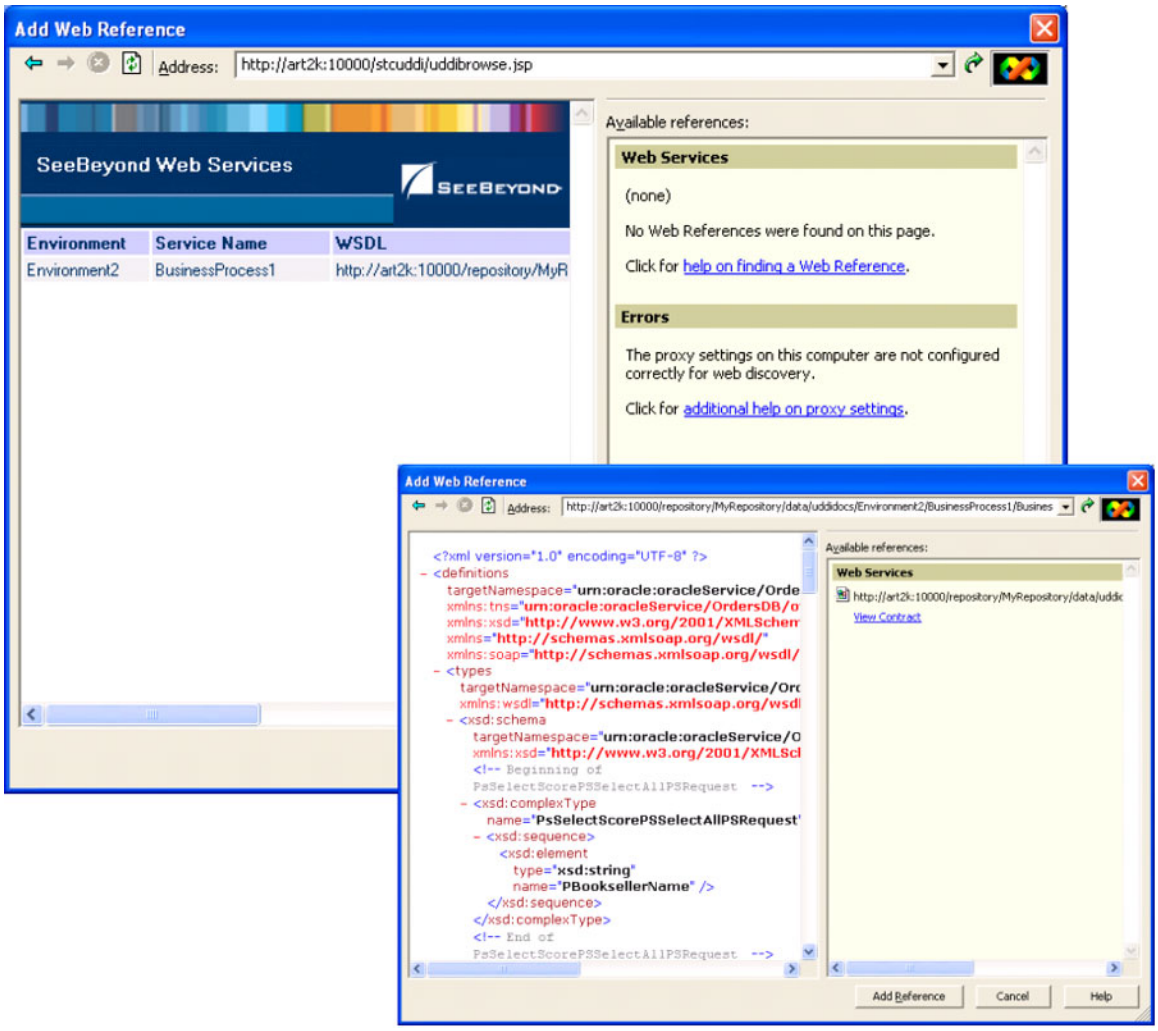
By selecting an entry its WSDL file is displayed, as shown in Figure 60.

Figure 60 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="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
  - <types targetNamespace="urn:oracle:oracleService/OrdersDB/otdGetCreditScore" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
    - <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" />
```

The SeeBeyond UDDI Repository can be used in a third party tool, for example Microsoft Visual Studio (see Figure 61). In this example, a so-called *Web reference* (to the UDDI repository) is added to a C# project.

Figure 61 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 **Using the WSDL Wizard** on page 111).

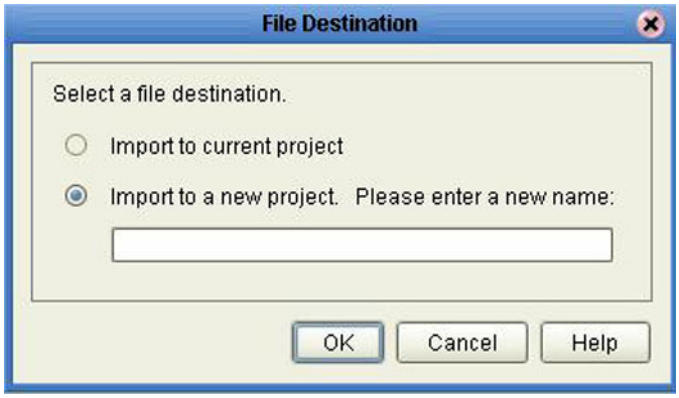
6.4 Building a Web Client

Here we briefly demonstrate the procedure for building a Web client. The Project used in the following example is available as **webclient.zip**, contained in the eGate ESB User Guide Sample file included with this User's Guide.

To import the sample project

1. The sample files are uploaded with the User's Guide **.sar** file and downloaded from the Enterprise Manager's Documentation tab. Extract the samples from the Enterprise Manager to a local file.
2. From the Enterprise Designer's Project Explorer pane, right-click the Repository and click **Import Project** form the selection menu. The **Select File to Import** dialog box appears.
3. Browse to the directory that contains the sample project zip file. Select the sample file (**webclient.zip**) and click **Open**.
4. From the *File Destination* dialog box (see Figure 62), select **Import to a new Project**, enter the name of the Project, and click **OK**.

Figure 62 File Destination Dialog Box



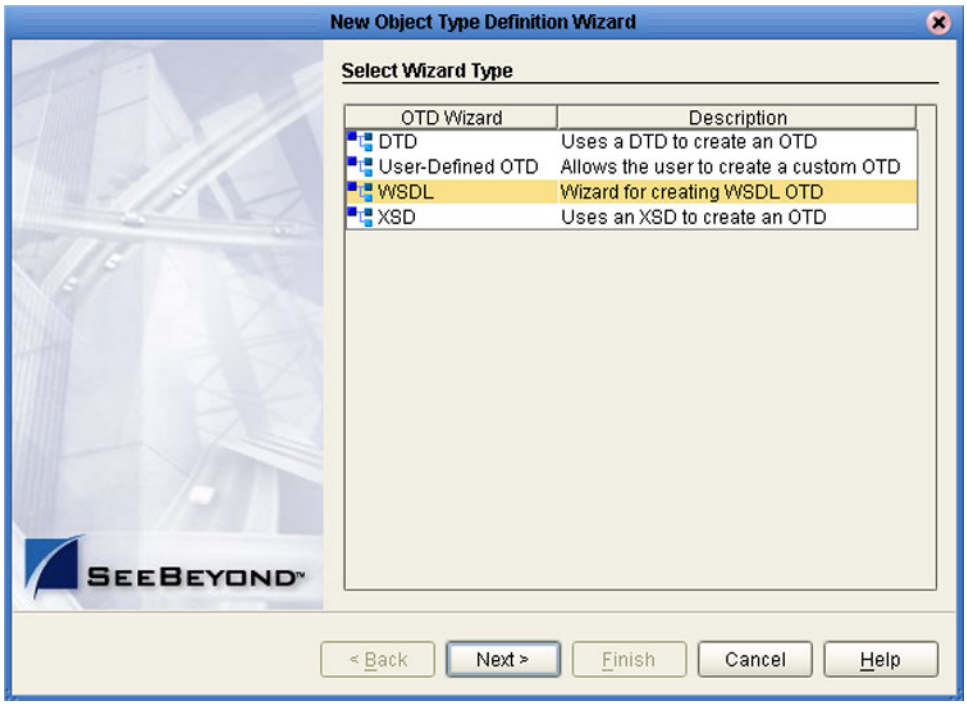
5. After the import has successfully completed, select the **Repository** in the Project Explorer and click **Refresh All from Repository**.

To build a Web Client Using the ICAN Suite

1 In Project Explorer, create a new OTD.

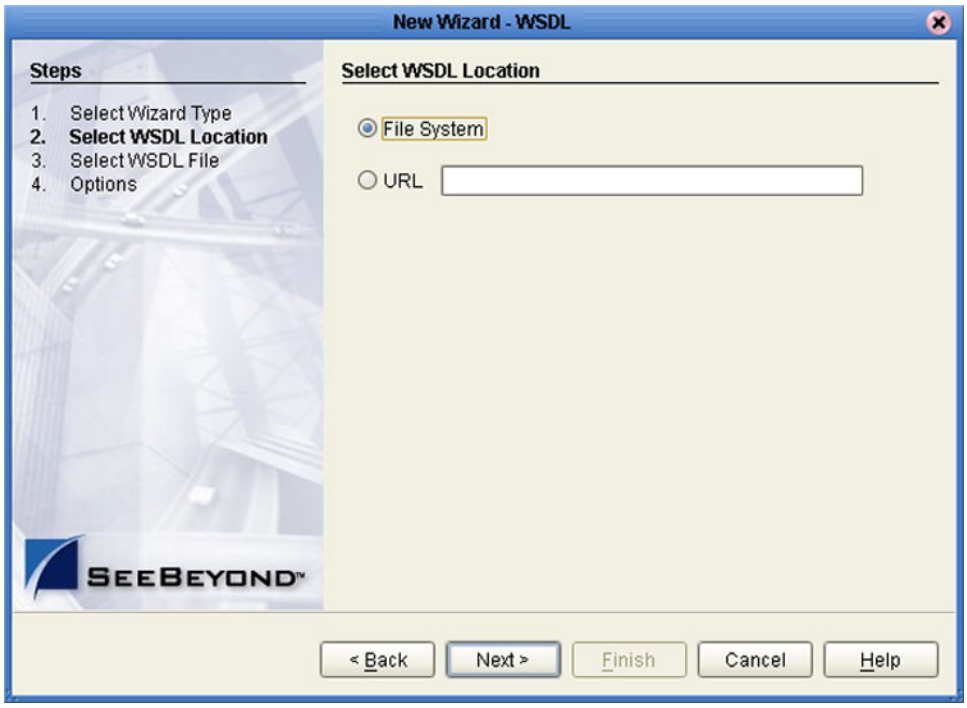
2 Select the WSDL OTD Wizard (see Figure 63).

Figure 63 Select WSDL Wizard



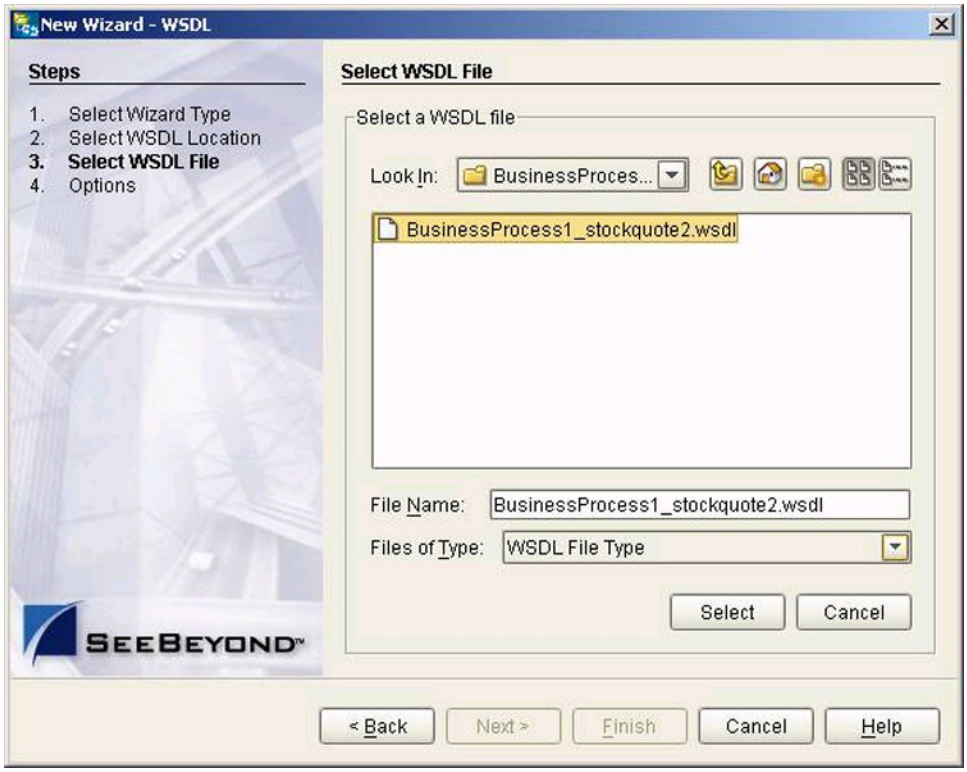
3 Select the WSDL file location (see Figure 64).

Figure 64 Select File Location



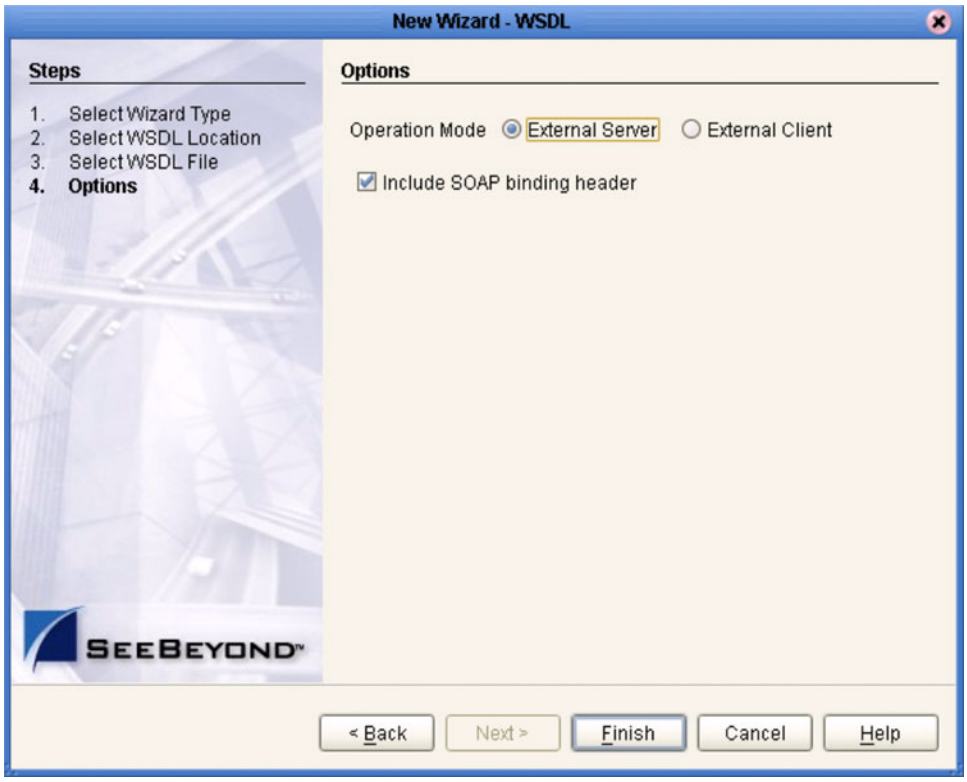
4 Select the WSDL file you want to use for the OTD (see Figure 65).

Figure 65 Select WSDL File



5 For a Web Client, select External Server (see Figure 66).

Figure 66 Select External Server



6 Build your business process using eInsight. The *receive* business rule is shown in Figure 67, and the *write* business rule is shown in Figure 68.

Figure 67 eInsight Business Process - Receive Rule

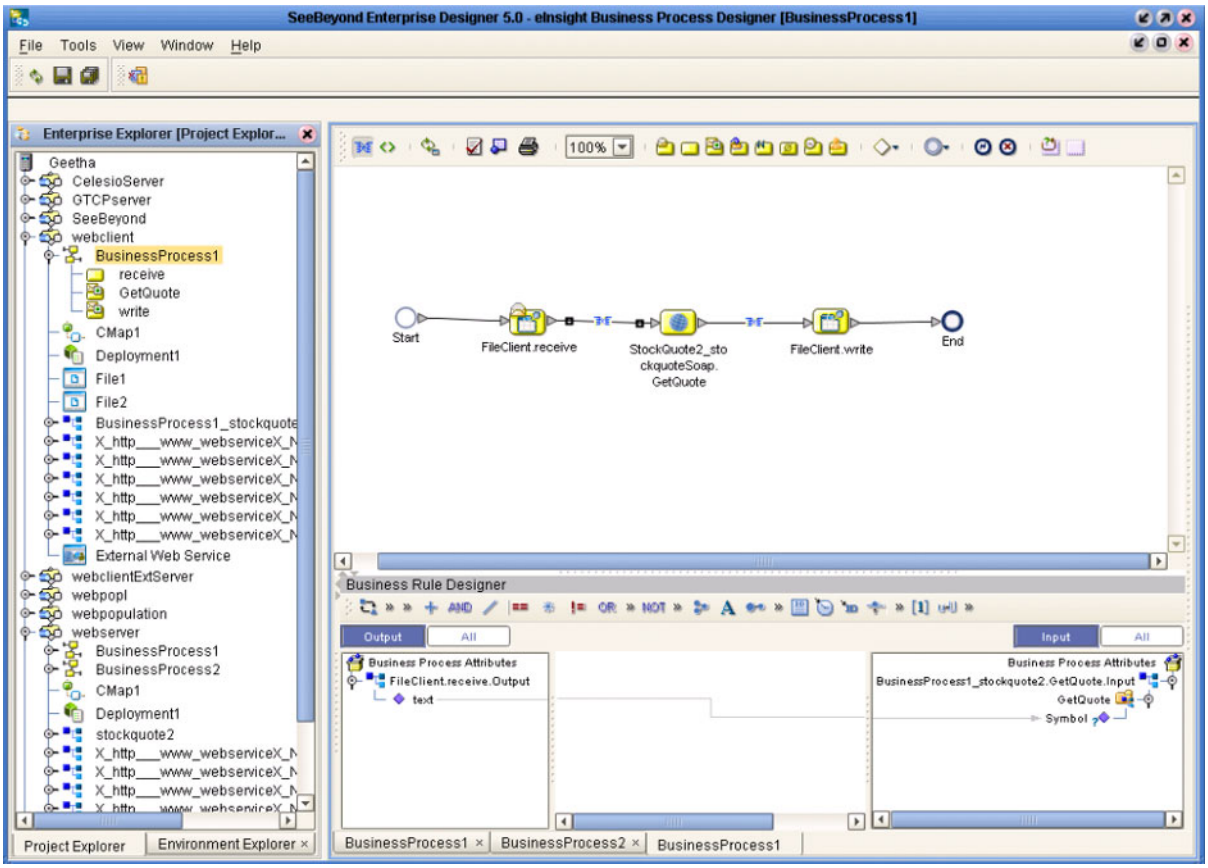
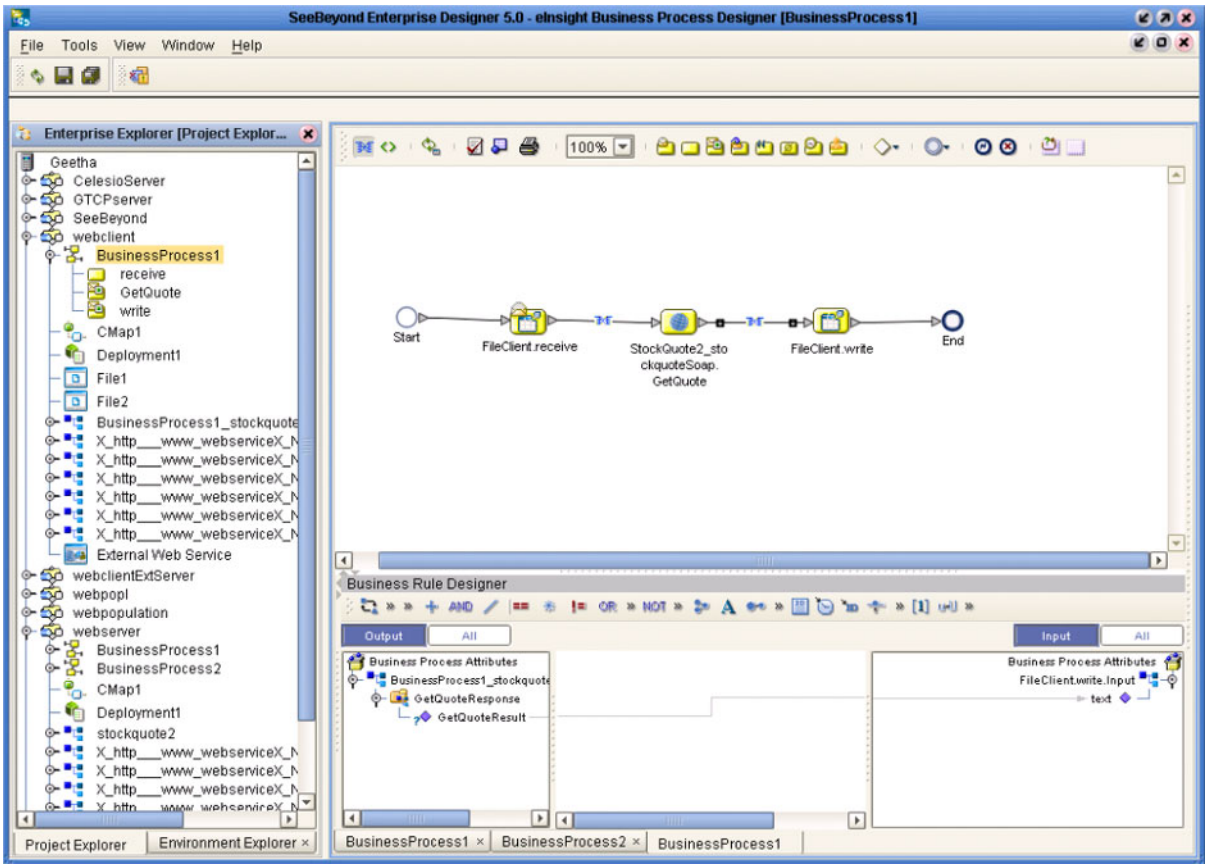
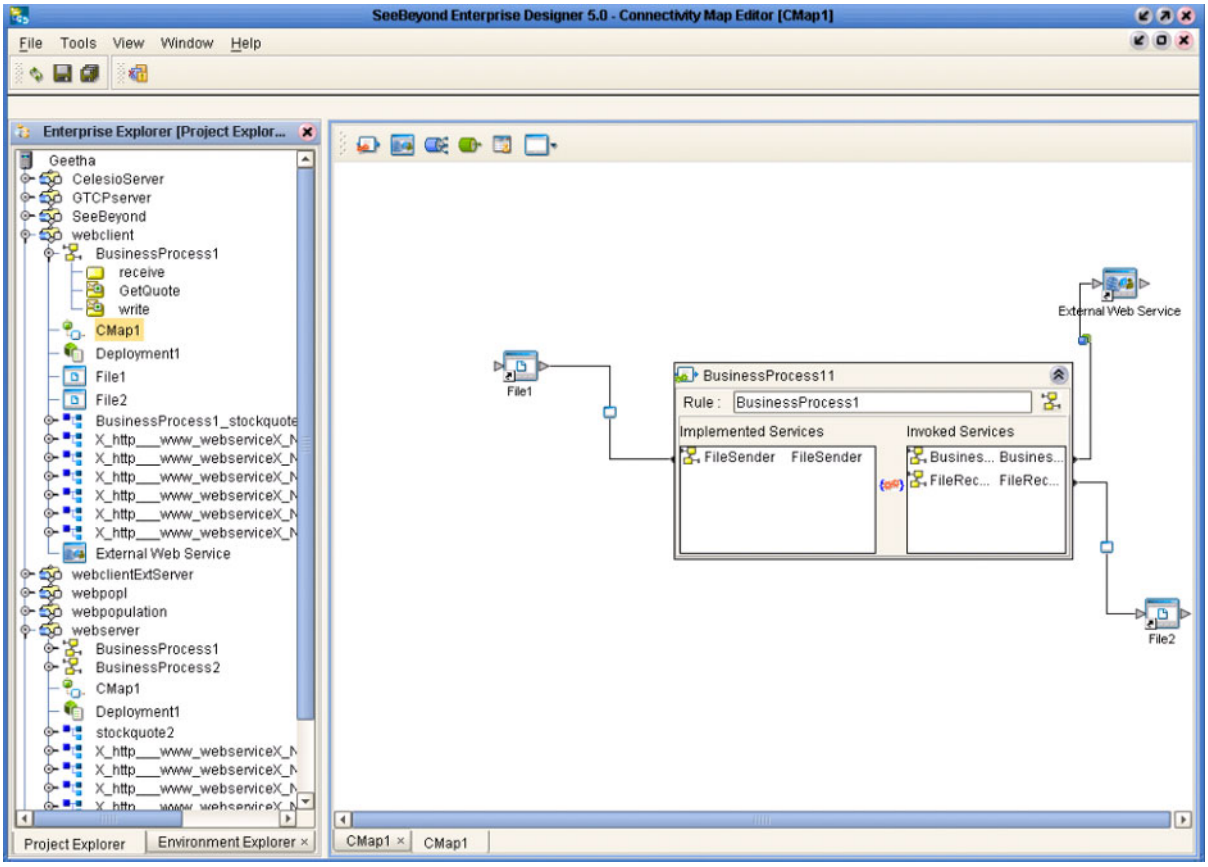


Figure 68 eInsight Business Process - Write Rule



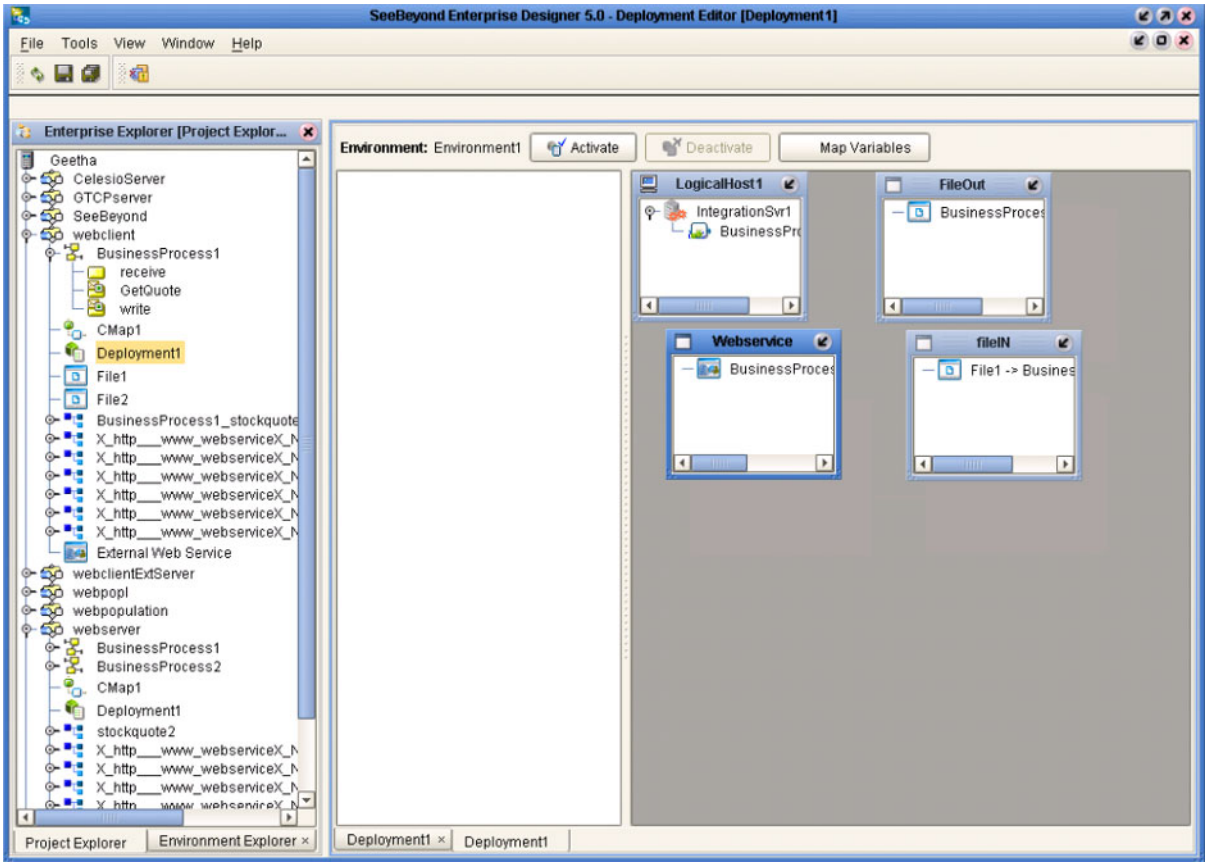
7 Create and map the Project components in the Enterprise Designer Connectivity Map (see Figure 69).

Figure 69 Connectivity Map



8 Deploy your Project as shown in Figure 70.

Figure 70 Project Deployment



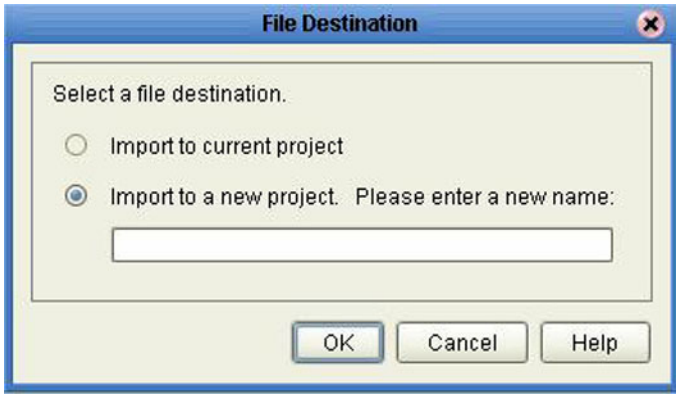
6.5 Building a Web Server

Here we briefly demonstrate the procedure for building a Web server. The Project used in the following example is available as **webserver.zip**, contained in the eGate ESB User Guide Sample file included with this User's Guide.

To import the sample project

1. The sample files are uploaded with the User's Guide **.sar** file and downloaded from the Enterprise Manager's Documentation tab. Extract the samples from the Enterprise Manager to a local file.
2. From the Enterprise Designer's Project Explorer pane, right-click the Repository and click **Import Project** form the selection menu. The **Select File to Import** dialog box appears.
3. Browse to the directory that contains the sample project zip file. Select the sample file (**webserver.zip**) and click **Open**.
4. From the *File Destination* dialog box (see Figure 62), select **Import to a new Project**, enter the name of the Project, and click **OK**.

Figure 71 File Destination Dialog Box



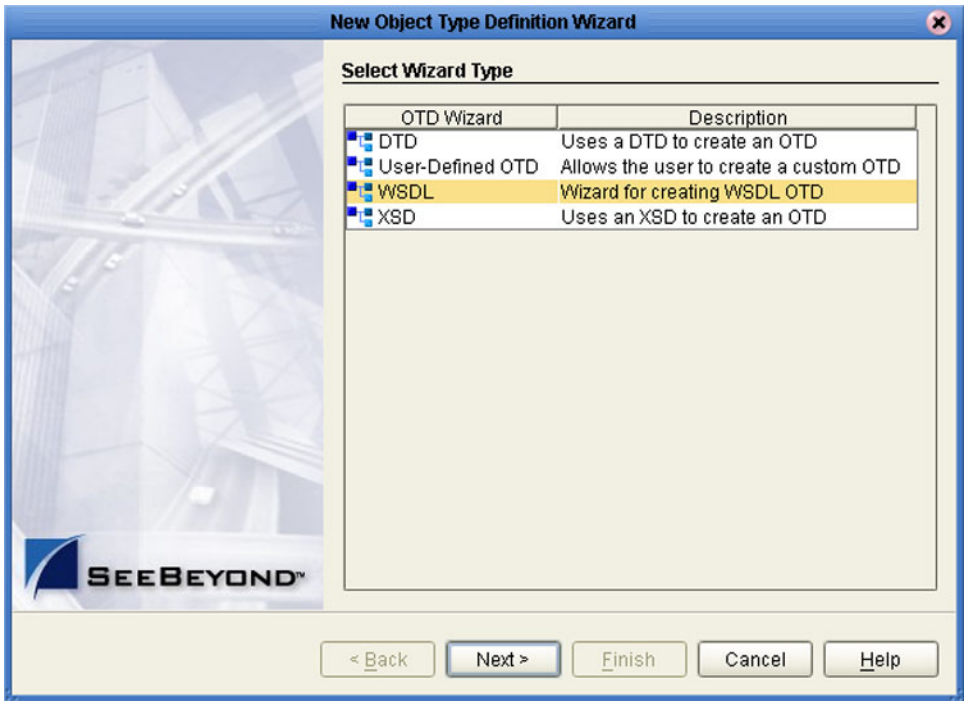
5. After the import has successfully completed, select the **Repository** in the Project Explorer and click **Refresh All from Repository**.

To build a Web Server Using the ICAN Suite

1 In Project Explorer, create a new OTD.

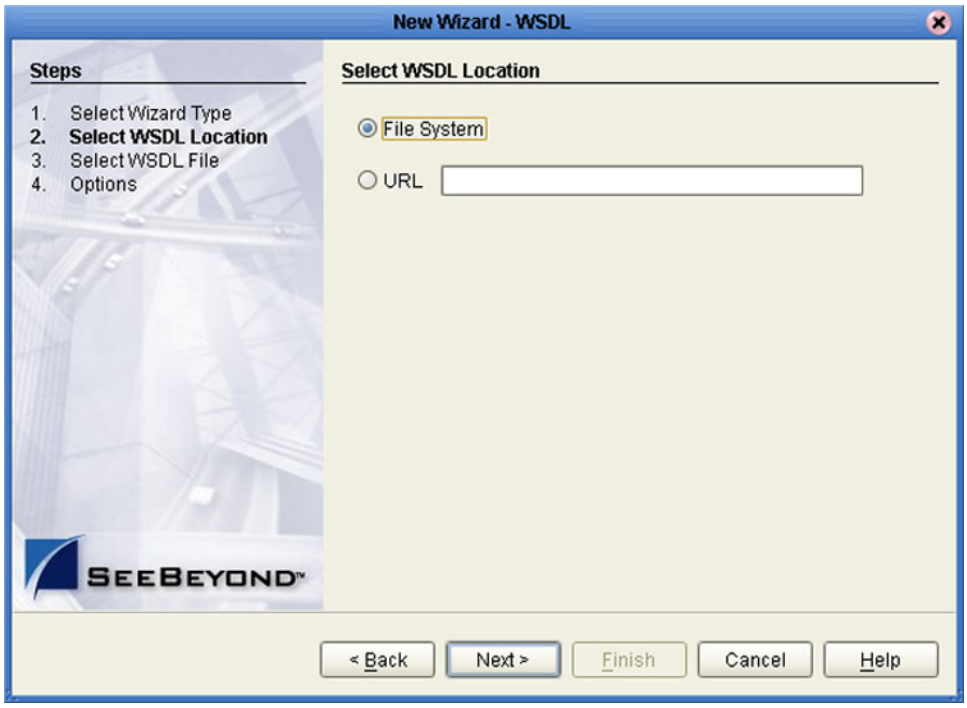
2 Select the WSDL OTD Wizard (see Figure 63).

Figure 72 Select WSDL Wizard



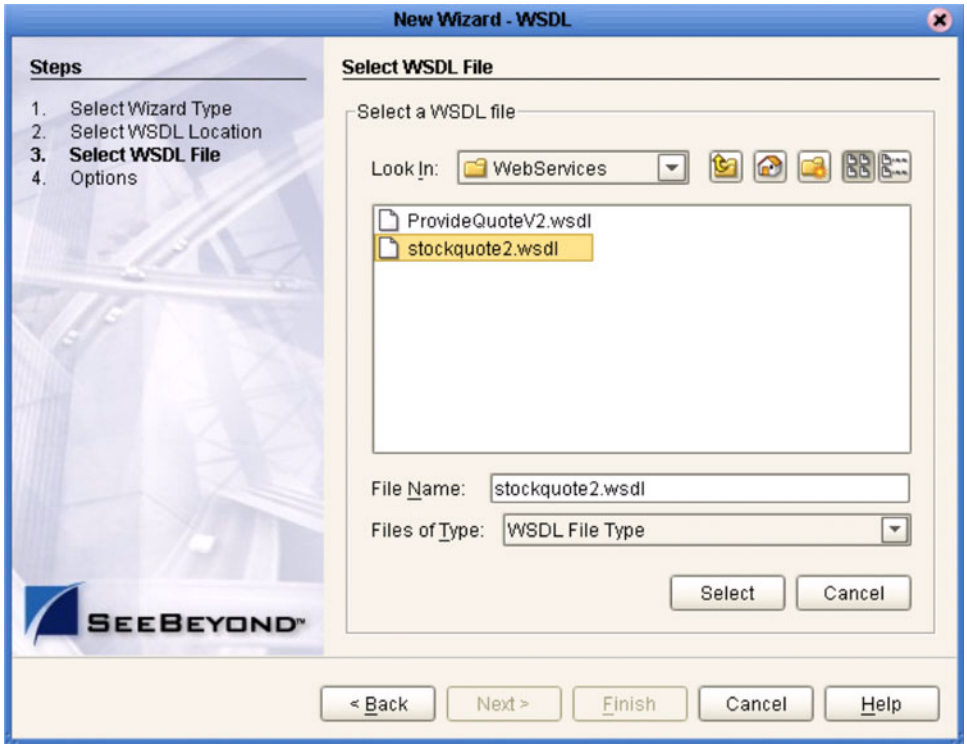
3 Select the WSDL file location (see Figure 73).

Figure 73 Select File Location



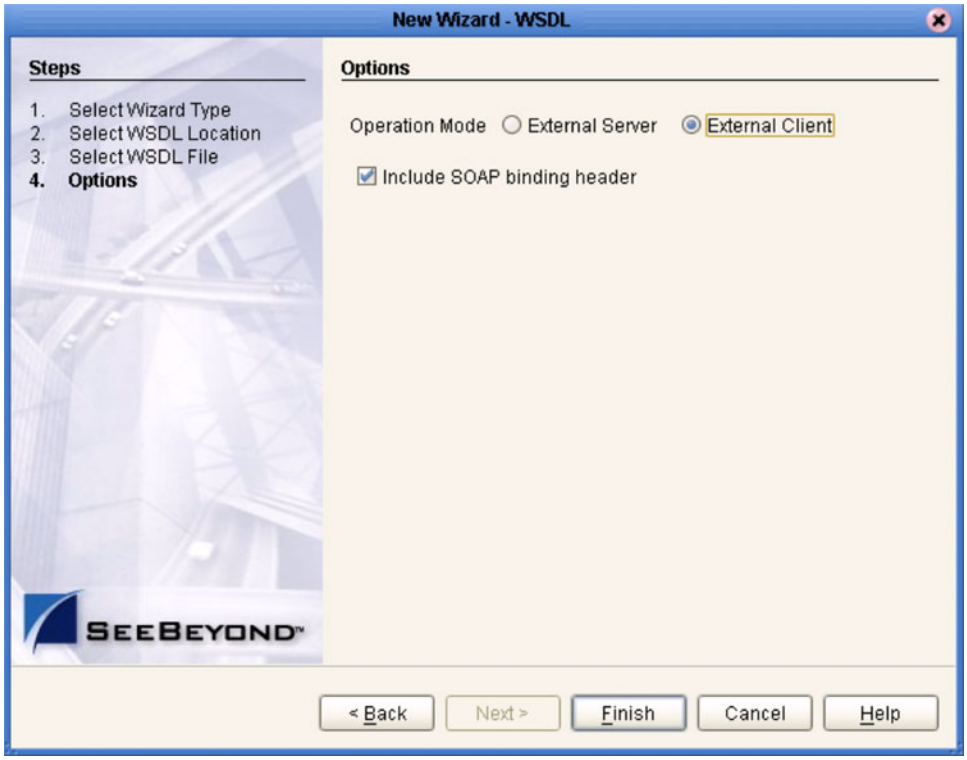
4 Select the WSDL file you want to use for the OTD (see Figure 74).

Figure 74 Select WSDL File



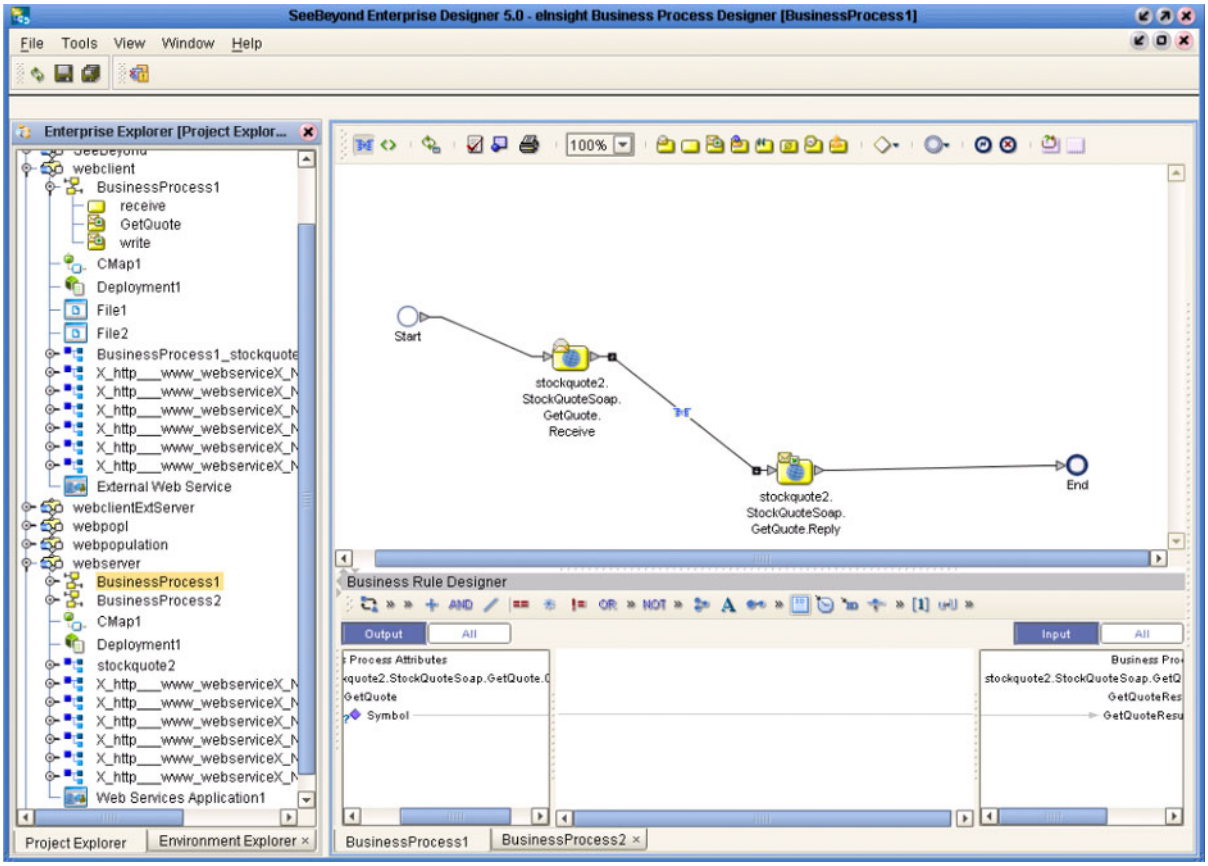
5 For a Web server, select External Client (see Figure 75).

Figure 75 Select External Client



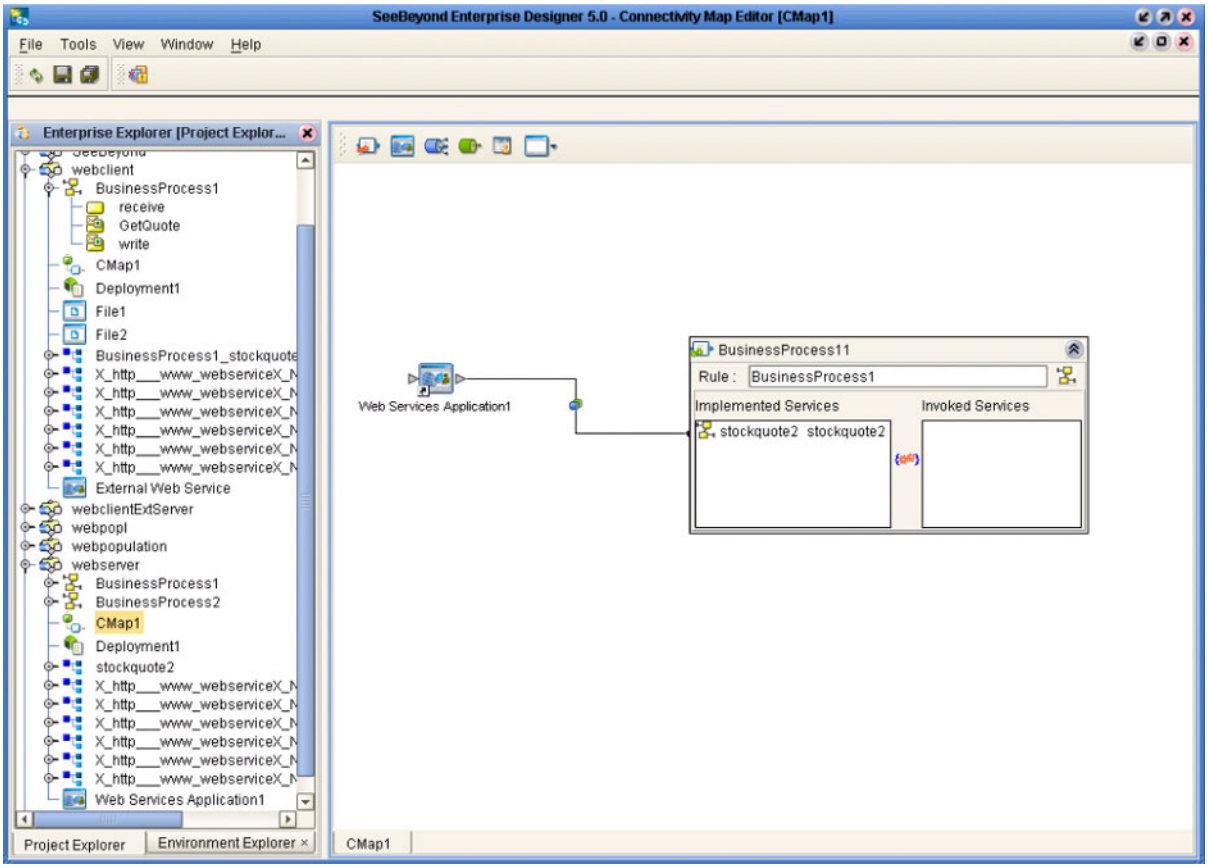
6 Build your business process using eInsight, as shown in Figure 76.

Figure 76 eInsight Business Process



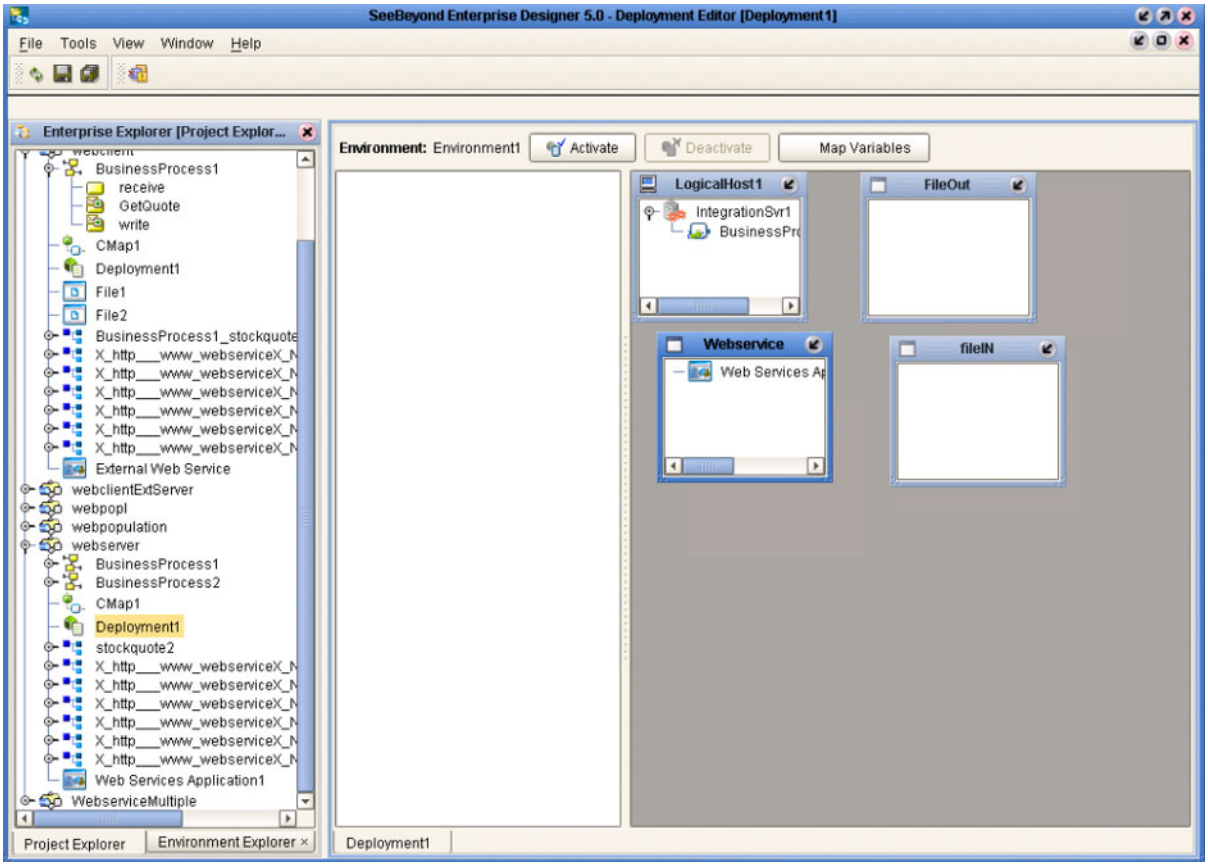
7 Create and map the Project components in the Enterprise Designer Connectivity Map (see Figure 69).

Figure 77 Connectivity Map



8 Deploy your Project as shown in Figure 78.

Figure 78 Project Deployment



Chapter 7

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.

7.1 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.

7.1.1 OTD Types

Externally-Defined OTDs

Externally-defined OTDs are based on formats or standards external to eGate Integrator, such as Document Type Definition (DTD), 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.

7.1.2 Building OTDs

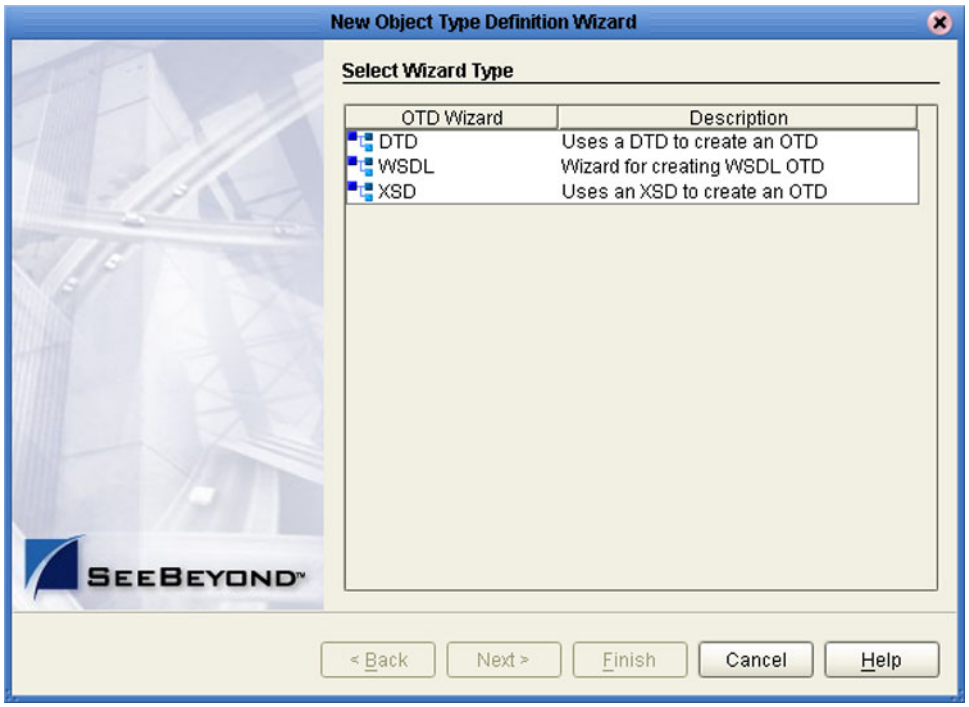
Wizards are provided in the Enterprise Designer GUI to guide you through the OTD building process. These Wizards call back-end builders that actually implement the building of the code, based on the provided information.

7.2 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 79. The initial dialog allows you to select a specific OTD Wizard. The basic Wizards supplied with eGate Integrator are described in:

- **Using the DTD Wizard** on page 107
- **Using the WSDL Wizard** on page 111
- **Using the XSD Wizard** on page 116

Figure 79 OTD Wizard Selection Dialog



Additional OTD Wizards are supplied with eGate Add-ons, and are described in the User's Guides for the specific products. When these products are installed, the OTD Wizards are added to the list shown in Figure 79.

The OTD Wizards guide you through the initial phases of creating an Object Type Definition, and then invoke the OTD Editor. The user interface is highly self-explanatory, but details of the navigation buttons are listed in Table 20 for your reference.

Table 20 OTD Wizard Navigation Buttons

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

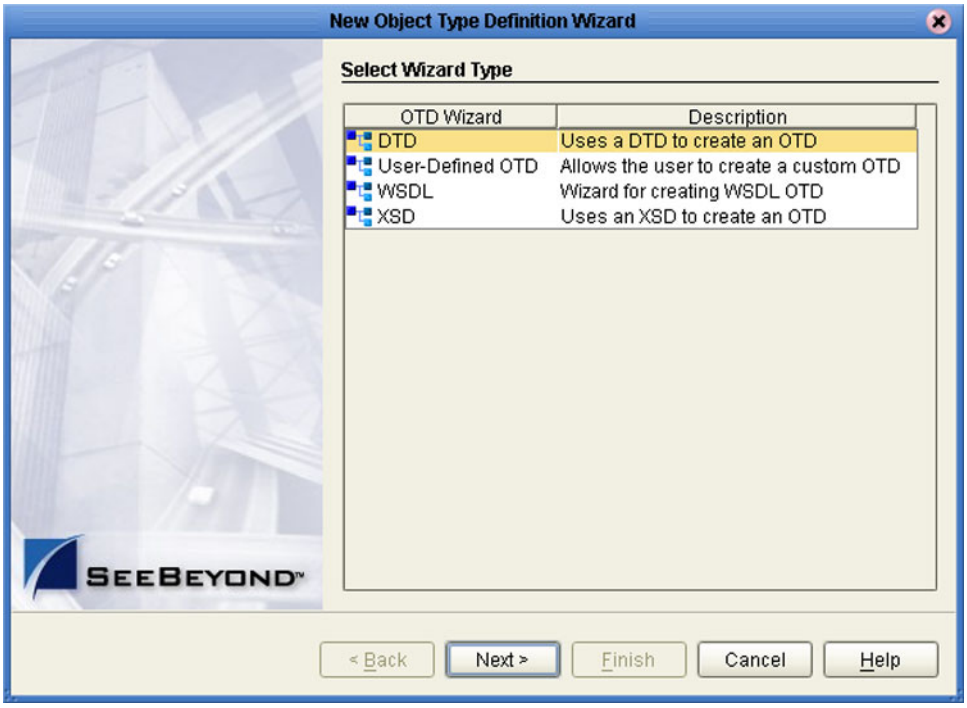
7.3 Externally-Defined OTDs

7.3.1 Using the DTD Wizard

To create an OTD file from a DTD file

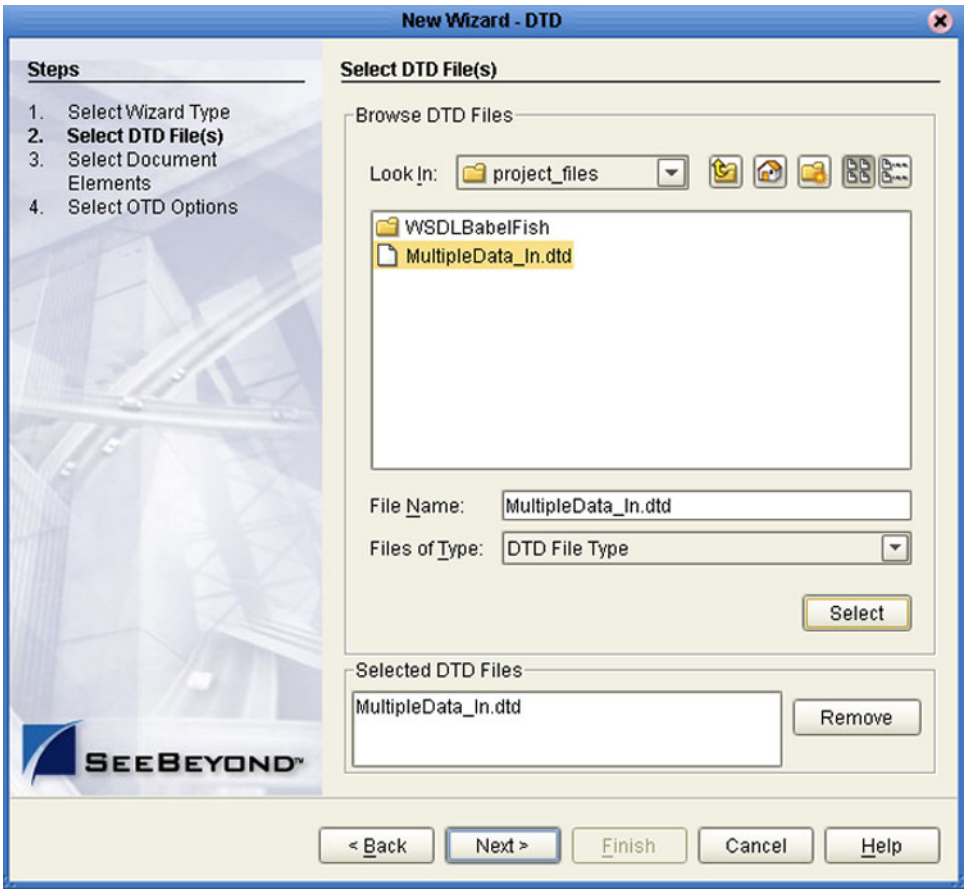
1 In the *Select Wizard Type* dialog, select **DTD** from the *OTD Wizard* list (see Figure 80) to create an OTD file from a Data Type Definition (DTD) file.

Figure 80 OTD Wizard Selection: DTD Wizard



2 Click **Next** to display the *Select DTD File(s)* dialog box, shown in Figure 81.

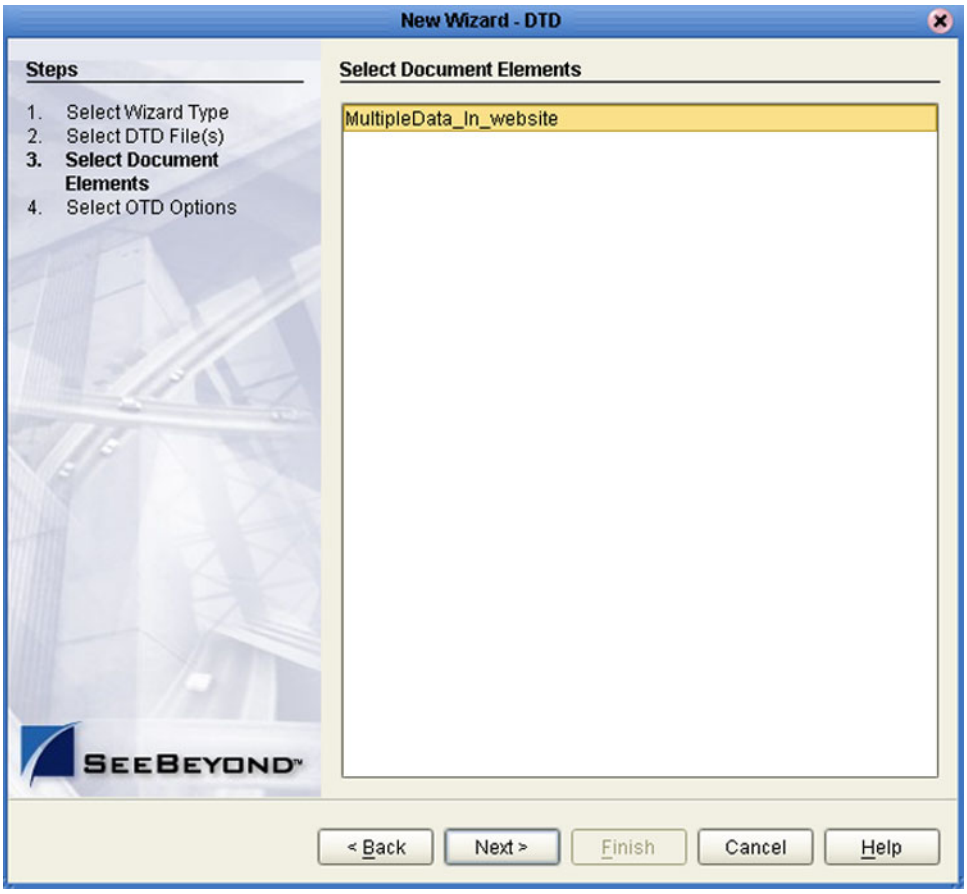
Figure 81 Select DTD File(s) Dialog Box



3 In the *Look In* drop-down list, navigate to the DTD file or files that you want to use to create the OTD. Click **Select** to add the files to the *List of Selected DTDs*.

4 Click **Next** to display the *Select Document Elements* dialog box, shown in Figure 82.

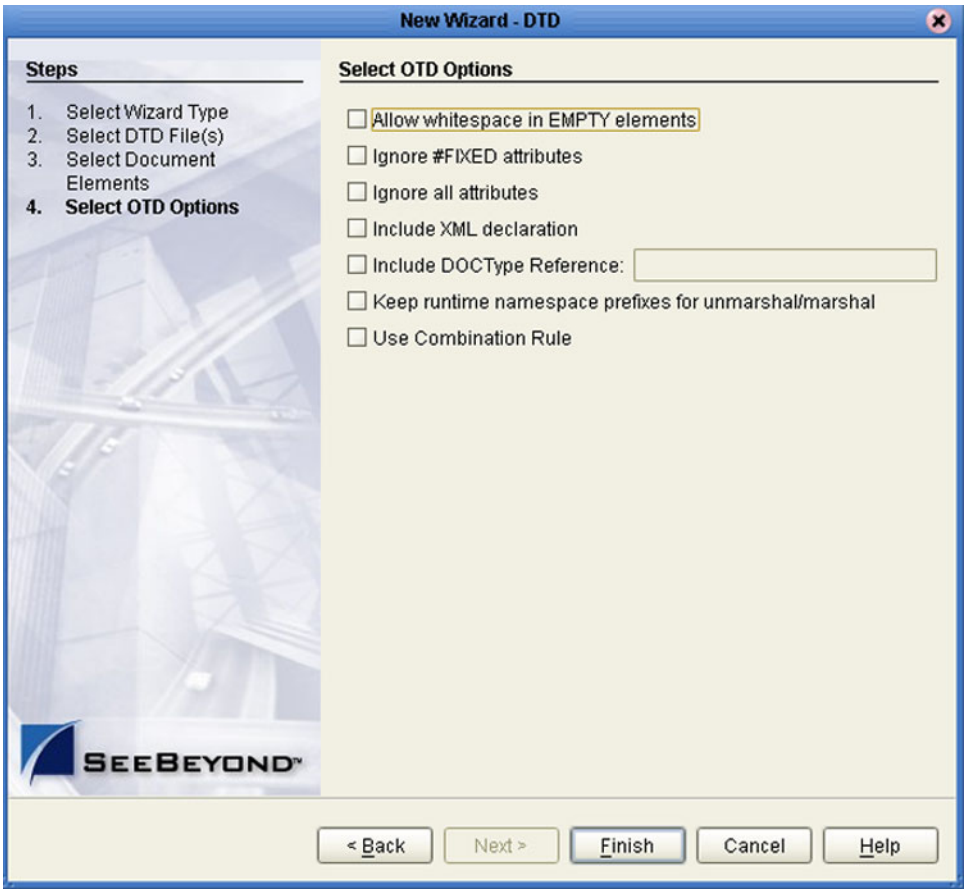
Figure 82 Select Document Elements Dialog Box



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 83.

Figure 83 Select OTD Options Dialog Box



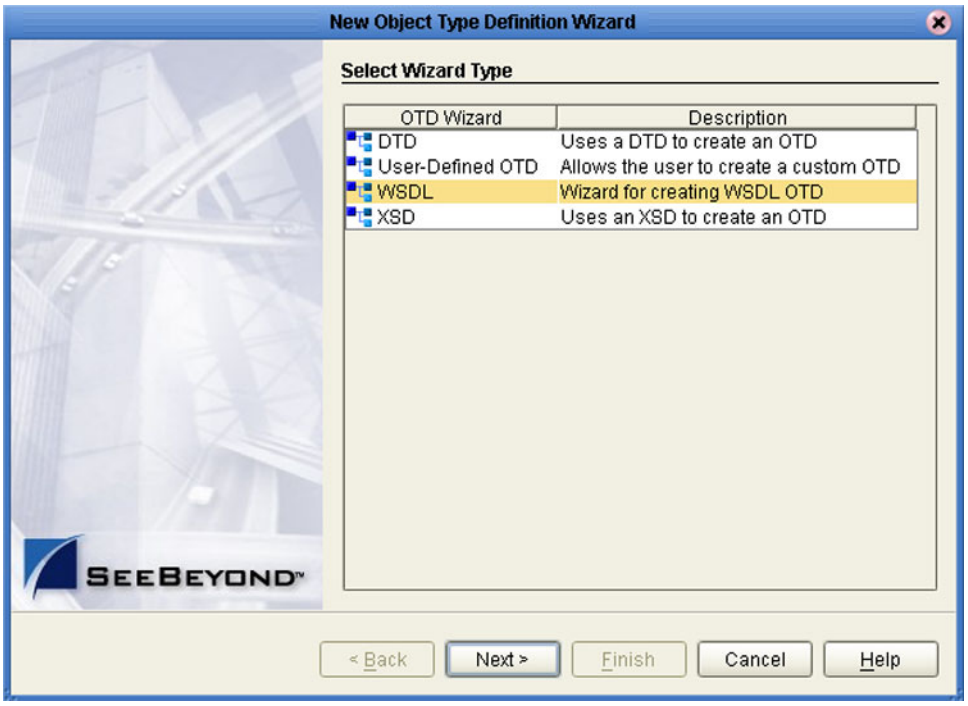
7 Select the check boxes next to the OTD options you want to enable, and click **Finish** to add the OTD to the Enterprise Designer with the selected OTD options.

7.3.2 Using the WSDL Wizard

To create an OTD file from a WSDL file

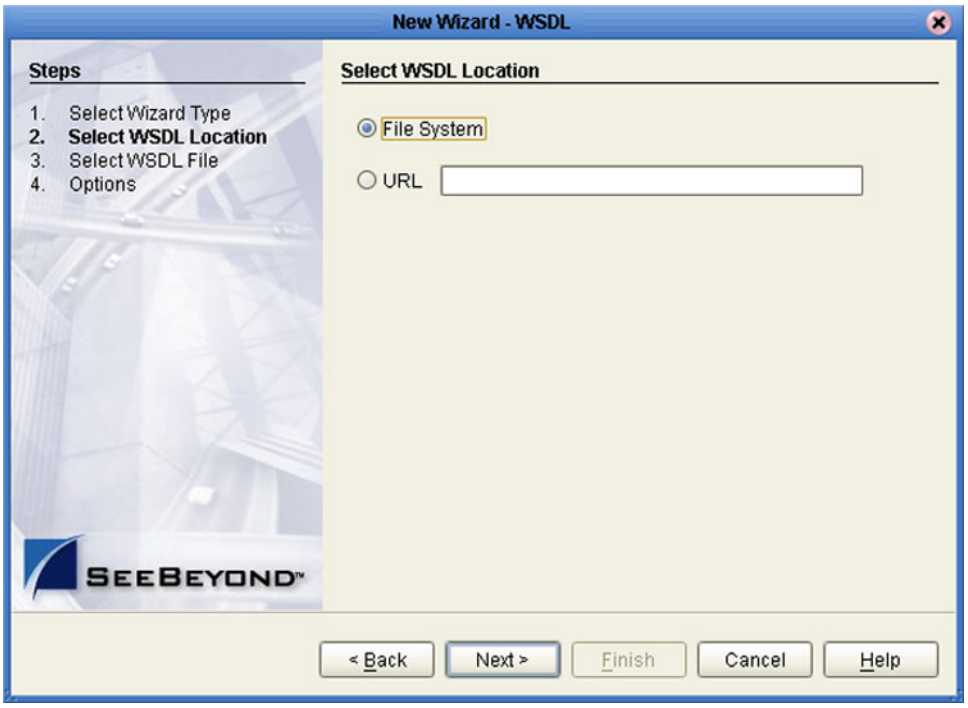
1 In the *Select Wizard Type* dialog, select **WSDL** from the *OTD Wizard* list (see Figure 84) to create an OTD file from an WSFL file.

Figure 84 OTD Wizard Selection: WSDL Wizard



2 Click **Next** to display the *Select WSDL File Location* dialog, shown in Figure 85

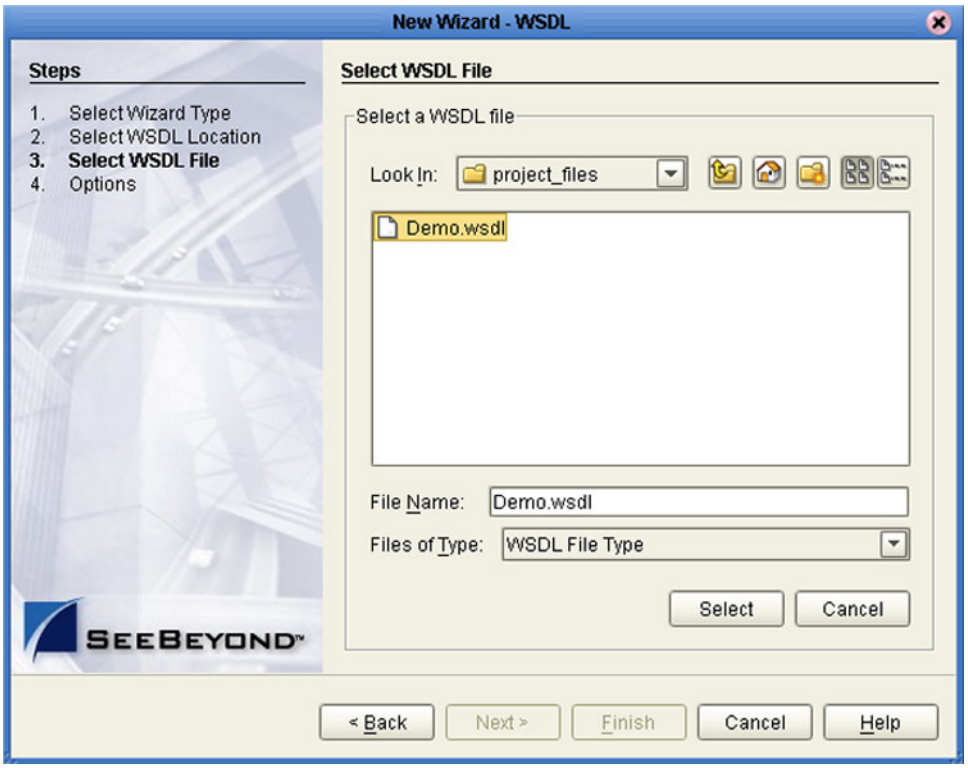
Figure 85 WSDL Wizard: Select WSDL Location



3 In the *Select WSDL Location* dialog, 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 86.

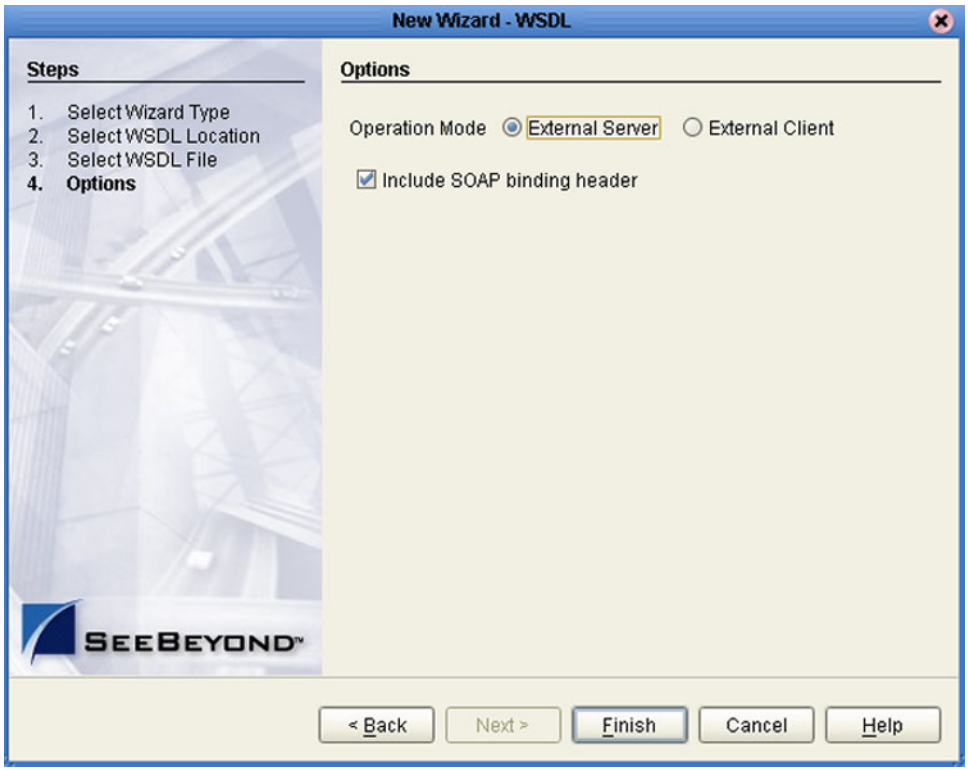
Figure 86 WSDL Wizard: Select WSDL File



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 *Options* dialog, shown in Figure 87.

Figure 87 WSDL Wizard: Select OTD Options



7 Select the check boxes next to the OTD options you want to enable:

- If you are using a Web client, select **External Server**.
- If you are using a Web server, select **External Client**.
- To include the SOAP binding header in the WSDL file, select the check box.

8 Click **Finish** to add the OTD to the Enterprise Designer with the selected OTD options.

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.

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:

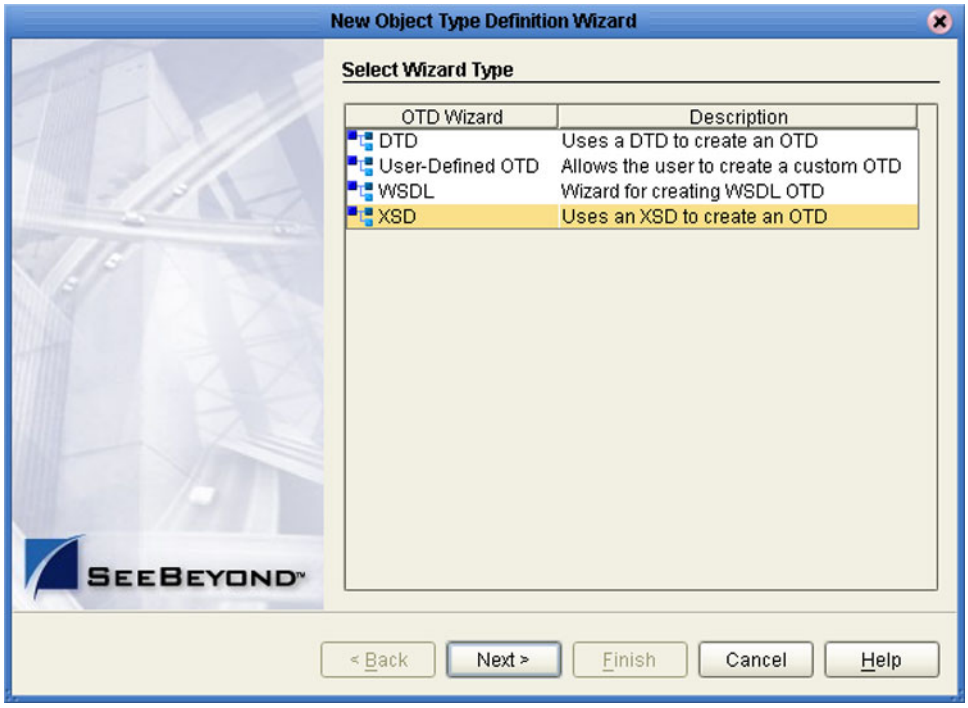
- **One-way:** The server receives a message from the client; also referred to as "fire and forget."
- **Request-response:** The server receives a message from the client and sends a correlated message back

7.3.3 Using the XSD Wizard

To create an OTD file from an XSD file

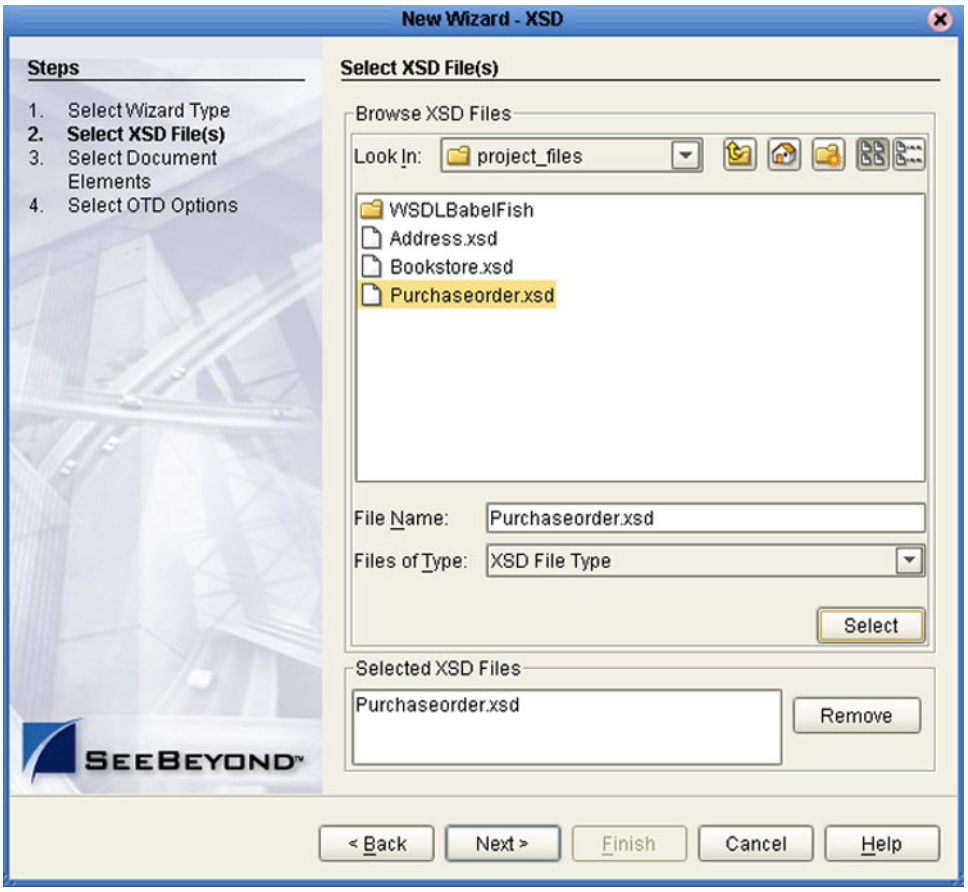
1 In the *Select Wizard Type* dialog, select **XSD** from the *OTD Wizard* list (see Figure 88) to create an OTD file from an XSD file.

Figure 88 OTD Wizard Selection: XSD Wizard



2 Click **Next** to display the Select XSD File(s) dialog box, shown in Figure 89.

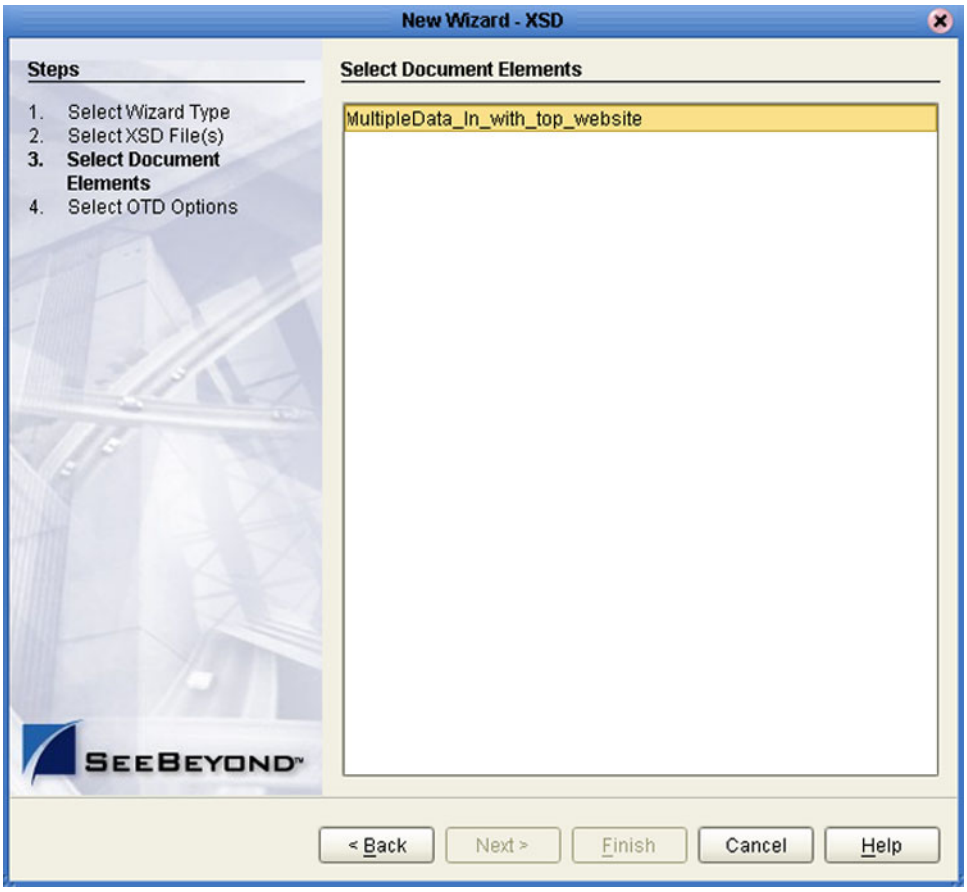
Figure 89 XSD Wizard: Select XSD File(s)



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 82.

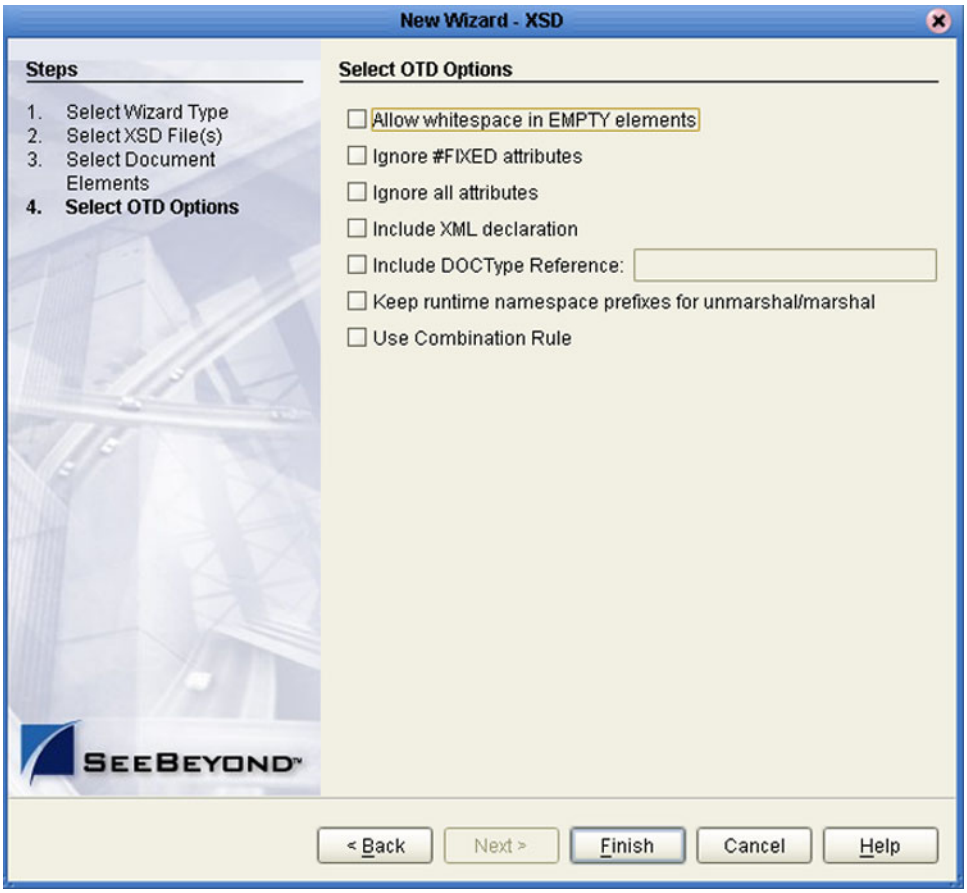
Figure 90 Select Document Elements Dialog Box



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 83.

Figure 91 Select OTD Options Dialog Box



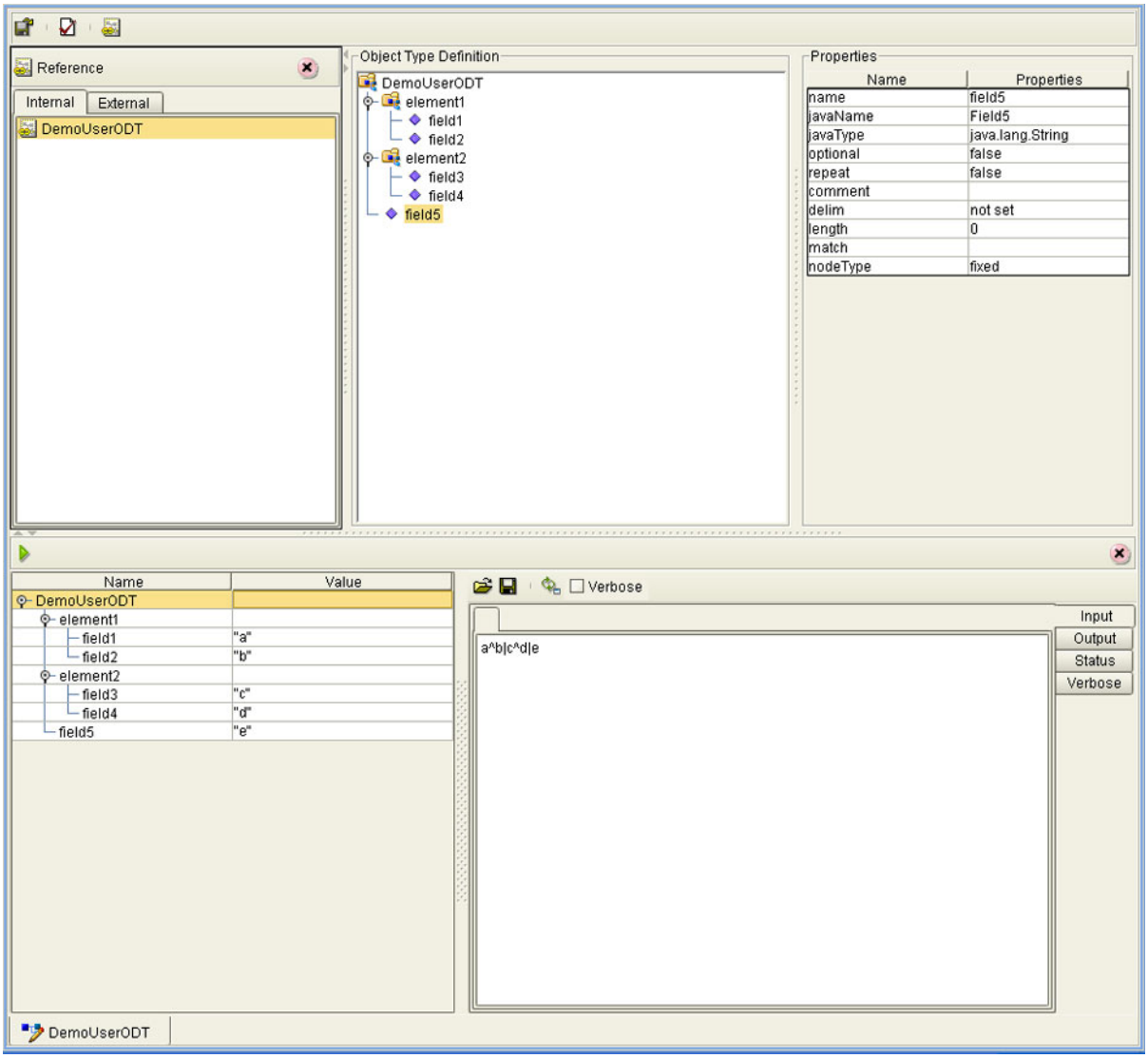
7 Select the check boxes next to the OTD options you want to enable, and click **Finish** to add the OTD to the Enterprise Designer with the selected OTD options.

7.4 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 92. 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 Analyzer** *on page 55**).*

Figure 92 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 21.

Table 21 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.

7.4.1 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.

7.5 Using the OTD Tester

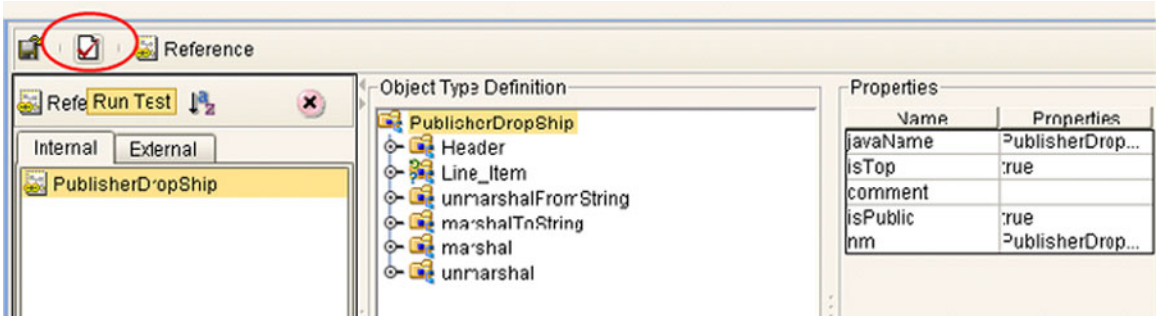
The OTD tester provides a facility to verify the correctness of OTDs, for example to:

- 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

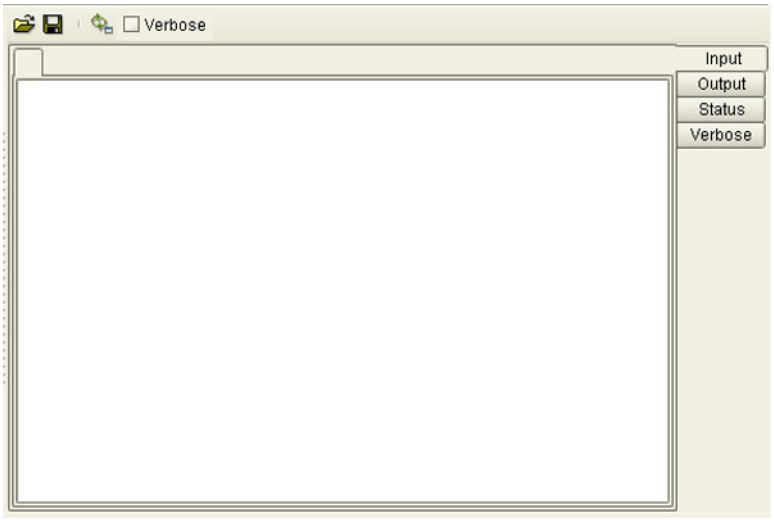
1. Open or create an OTD.
2. Click the **Tester** icon (see Figure 93).

Figure 93 OTD Tester



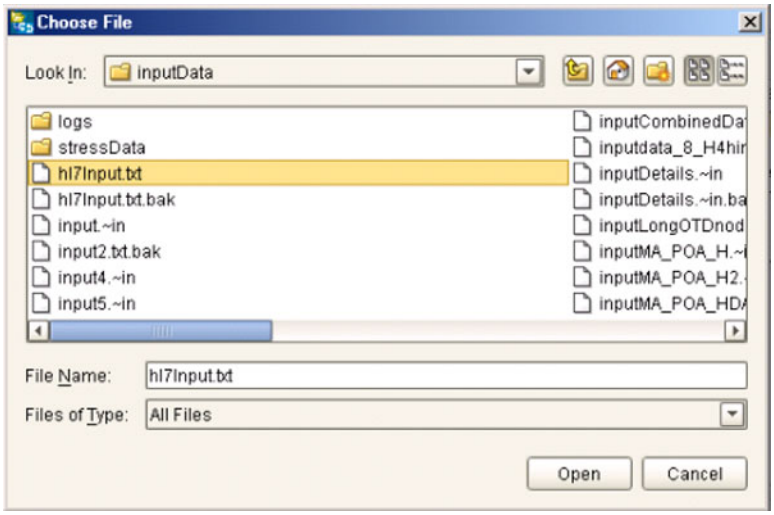
A test panel will appear below the OTD detail area of the editor. Note that there are four data display modes, selectable by tabs (see Figure 94). The Input tab is selected by default.

Figure 94 Test Panel Data Display



3 You can provide the input test data either by selecting a data file (see Figure 95), or by entering the data manually.

Figure 95 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 96).

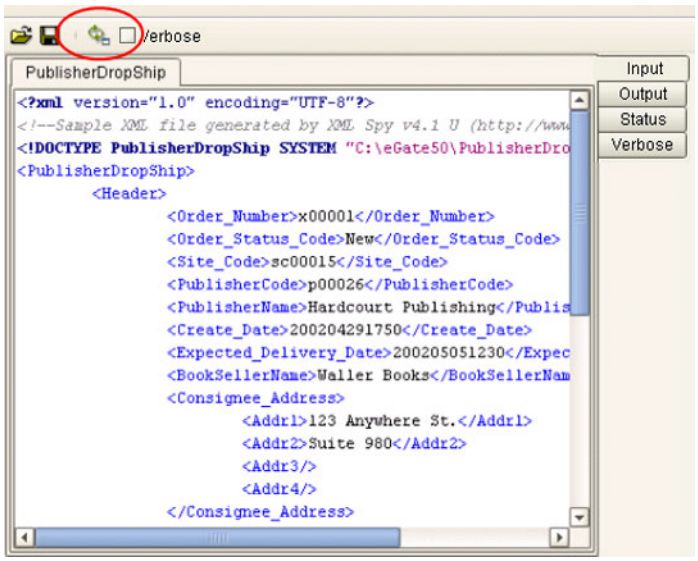
Figure 96 Object Elements and Values

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.

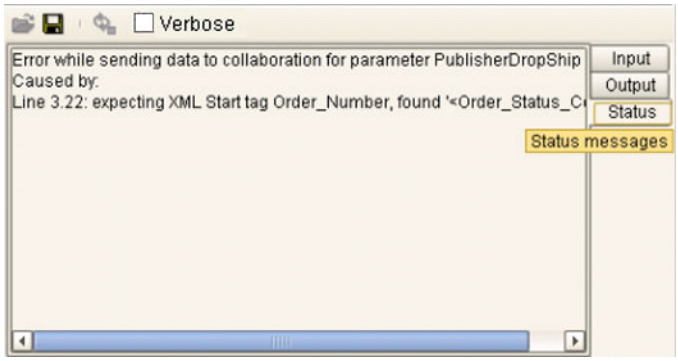
7 You can also change your test data in the Input data display, then re-test the OTD by clicking the **Refresh** icon (see Figure 97) to repopulate your OTD object elements with the new values.

Figure 97 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 98).

Figure 98 Status Data Display



Chapter 8

Environments

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. This chapter describes the process of defining eGate Environments, and the components of an Environment.

8.1 Overview

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

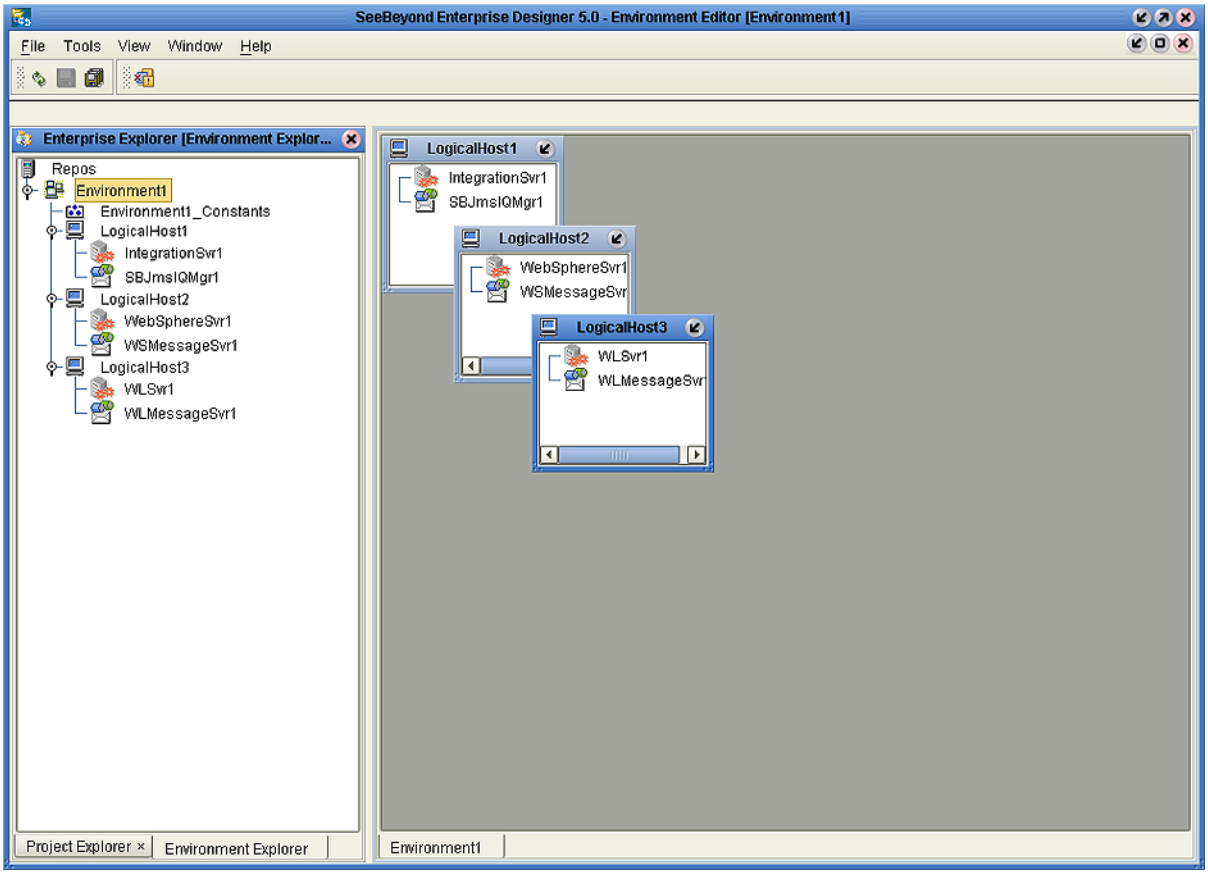
8.2 Environment Explorer

Figure 99 Enterprise Explorer: Environment Explorer View

8.3 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 (see Figure 100).

Figure 100 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 **Repository Menu** on page 129). Components are added to the Environment by selecting options in the Environment and Logical Host context menus (see **Environment Menu** on page 130 and **Logical Host Menu** on page 131, respectively).

8.3.1 Environment Explorer Icons

The icons described in Table 22 appear in the Environment Explorer.

Table 22 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.

8.3.2 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 129
- **Environment Menu** on page 130
- **Logical Host Menu** on page 131

Repository Menu

Figure 101 Repository Menu

Table 23 Repository Menu Options

Option	Function
New Environment	Displays a dialog box with which you can create a new Environment.
Configure SNMP Agent	Displays a dialog box in which you can modify the SNMP agent properties.
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 102 Environment Menu

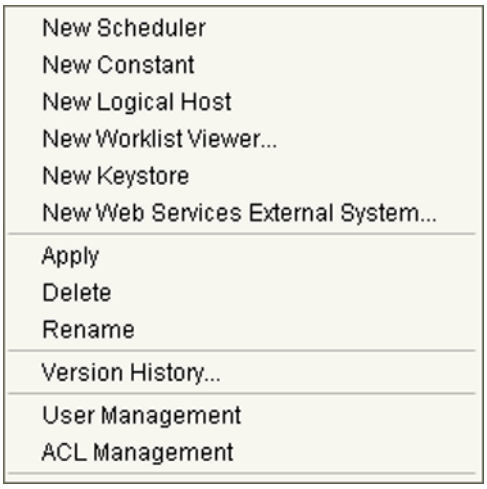


Table 24 Environment Menu Options

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. See **Defining Environmental Constants** on page 133.
New Logical Host	Adds a new Logical Host to the selected Environment.
New Worklist Viewer	See the *eInsight Enterprise Service Bus User's Guide* for information.
New Keystore	Adds a new keystore to the selected Environment.
New Web Service ...	Adds a third-party Web service application to the Project Explorer. See **SeeBeyond Web Services** on page 86.
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 **Viewing a Component's Version History** on page 59 for more information.
User Management	Displays a dialog box with which you can manage Repository access. See **Configuration User Management** on page 163.
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 190.

Logical Host Menu

Figure 103 Logical Host Menu

Table 25 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 **Viewing a Component's Version History** on page 59 for more information.
Check In	Displays a dialog box, with which you can check in the current version of an Environment. Refer to **Checking a Component In** on page 57 for more details.
Check Out	Displays a dialog box with which you can check out a version of an Environment. See **Checking a Component Out** on page 58 for more information.

Table 25 Logical Host Menu Options

Option	Function
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 190.
Properties	Displays a dialog box with which you can modify the default settings for the selected Logical Host.

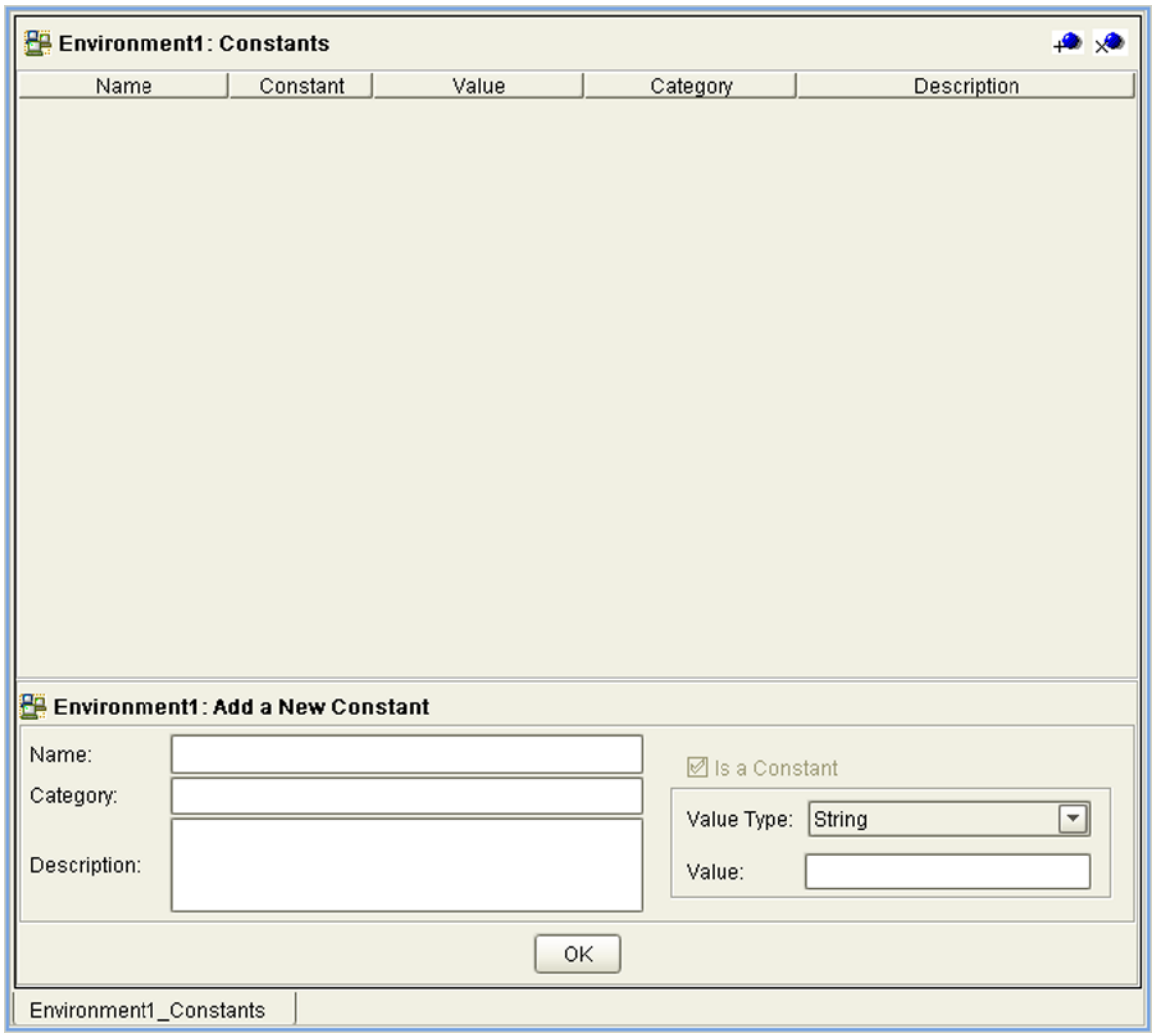
Note: *If you are using BEA WebLogic and/or IBM WebSphere, the Integration Servers and JMS Message Servers for these products will also appear in the context menu.*

8.4 Defining 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 104).

Figure 104 Environmental Constants Editor



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.

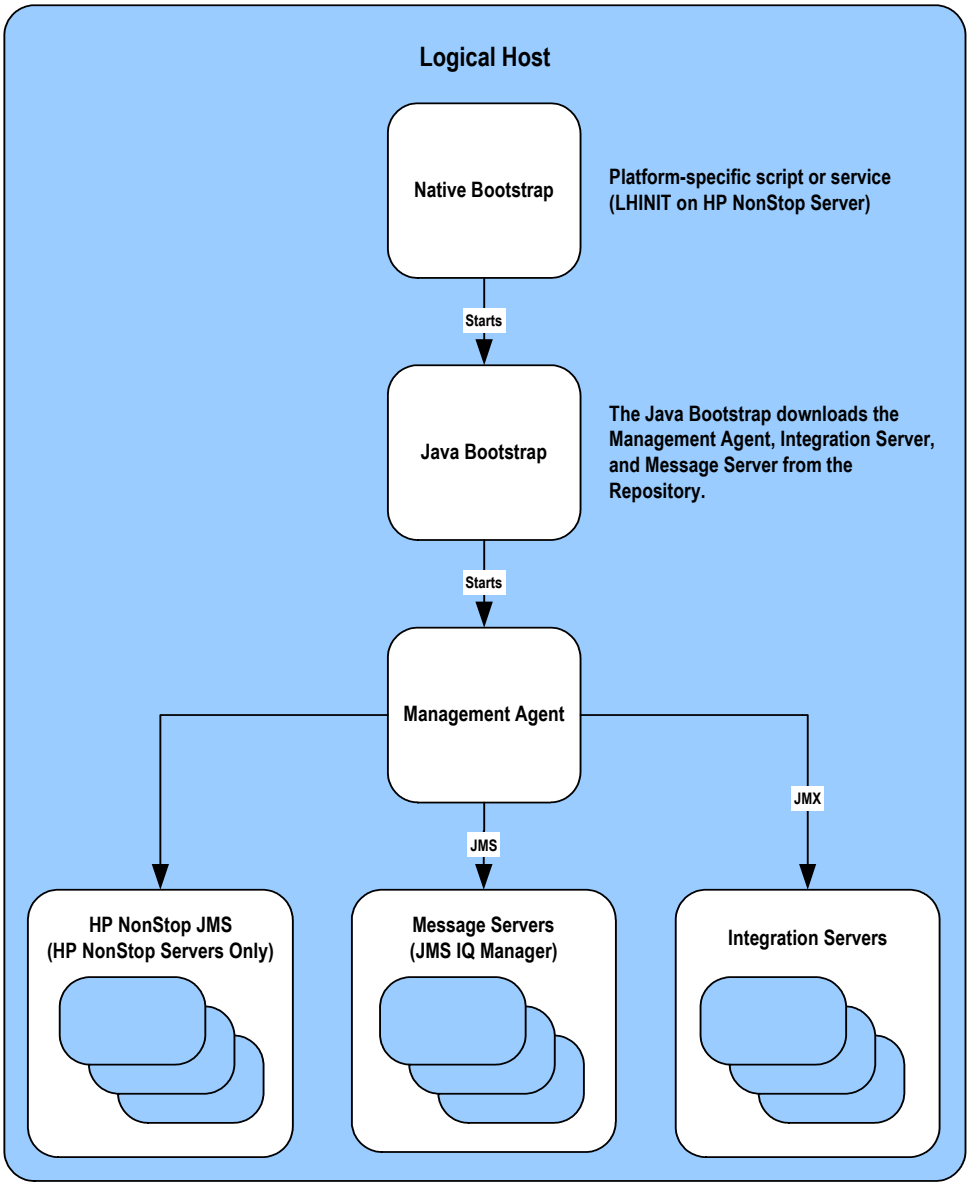
Table 26 Environmental Constants Editor Icons

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.

8.5 Logical Hosts

A Logical Host is an instance of the eGate runtime environment that is installed on a host hardware platform. The architecture of a Logical Host is illustrated in **Figure 105 on page 135**. A Logical Host can be a member of only one Environment, but each Environment can contain multiple Logical Hosts.

Figure 105 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).

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 144.

Note: *When multiple Logical Hosts reside on a single hardware platform, you must configure the base port numbers (see* **Configuring the Base Port Number** *on page 143).*

8.5.1 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.

8.5.2 Message Servers

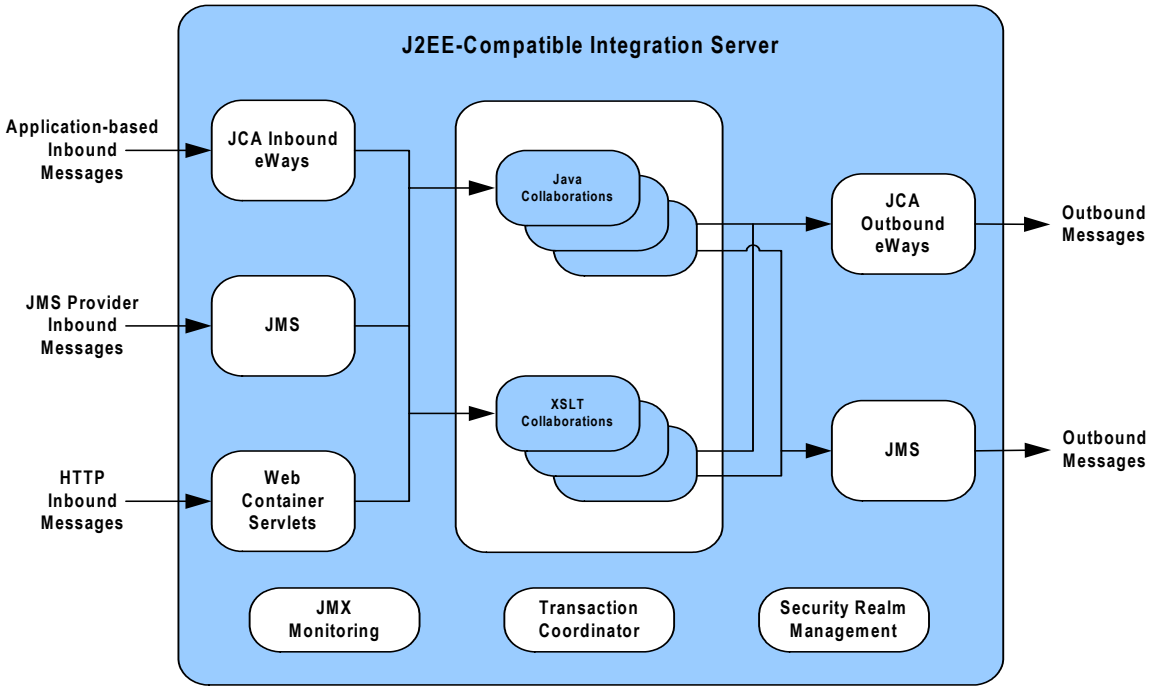
The Logical Host contains one or more Message Servers, which manage JMS topics (publish-and-subscribe messaging) and queues (point-to-point messaging). eGate Integrator includes the SeeBeyond JMS IQ Manager as its Java Messaging Service (JMS) implementation. The JMS IQ Manager conforms to the Java Message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

8.5.3 Integration Servers

The Logical Host contains one or more Integration Servers, which are the engines that run 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 155).

Figure 106 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.

8.6 Configuring the Logical Host

8.6.1 Modifying the Logical Host Properties File

If you want to install a Logical Host as a Windows service, you must set the Logical Host default startup configuration in the **logical-host.propterties** file. Setting the default configuration also avoids having to specify the arguments if you choose to start the Logical Host manually at the command prompt. If you do specify the arguments when you manually start the Logical Host, what you enter will override the default values; see **Starting the Logical Host** on page 144.

To modify the Logical Host Properties file

1. Ensure that the Logical Host is not running.
2. Navigate to *ICAN_root*\logicalhost\bootstrap\config\logical-host.properties.
3. Use a text editor (such as Windows WordPad) to open the file.

Figure 107 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\\stcjms.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
```

4 In the beginning of the **logical-host.properties** file, locate the properties listed in **Table 27 on page 140** (most are marked USER MODIFIABLE, and may appear in a different order than that shown):

5 Enter the appropriate values for your system.

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

Table 27 User-Modifiable Logical Host Properties

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:\ican50\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 path to your Repository; for example, **http://*hostname*:*port*/*repositoryname*** where: ▪ ***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. *Do not enter anything for this property or modify its contents*.

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

7 The Logical Host may now be started, either manually or as a Windows service.

8.6.2 Installing the Logical Host as a Windows Service

Installing the Logical Host as a Windows service configures the Logical Host to start up automatically at system startup, and restart automatically after an abnormal system shutdown.

Note: *You must have Administrator privileges on the local Windows computer in order to configure the Logical Host to start as a service (the installation script writes to the Windows registry, which requires Administrator privileges).*

To install the Logical Host as a Windows Service

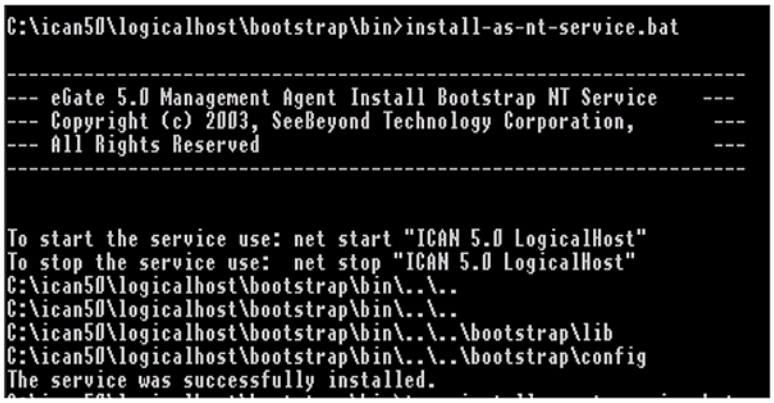
1. Configure the **logical-host.properties** file as described in **Modifying the Logical Host Properties File** on page 138.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root\logicalhost\bootstrap\bin\
   ```

3. Double-click **install-as-nt-service.bat**.

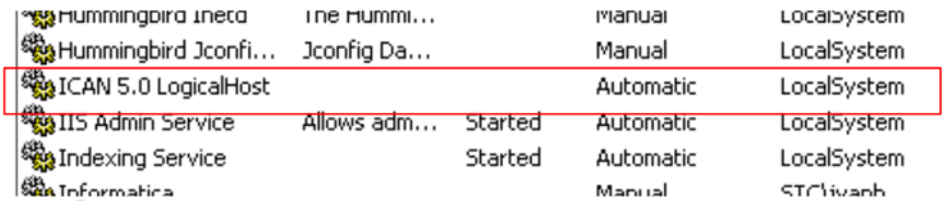
   This runs the script that configures the Windows service to automatically run the Logical Host at system startup. The only parameter is the name of your Logical Host Service; the default is "ICAN 5.0 Logical Host" (see Figure 108).

Figure 108 Install as Service Script



4. Verify the installation by opening the Windows Services facility and searching for the Logical Host name (see Figure 109). 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 109 Windows Services List



To remove the Logical Host Windows Service

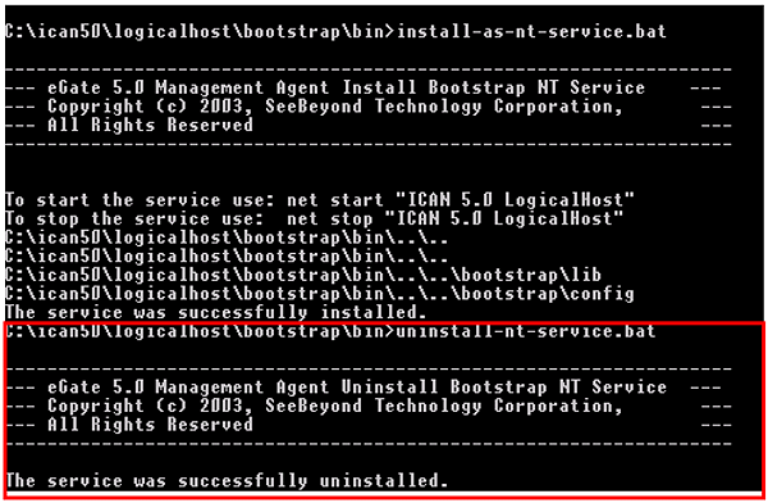
1 Change to the directory containing **bootstrap.bat**:

```
cd ICAN_root\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 110).

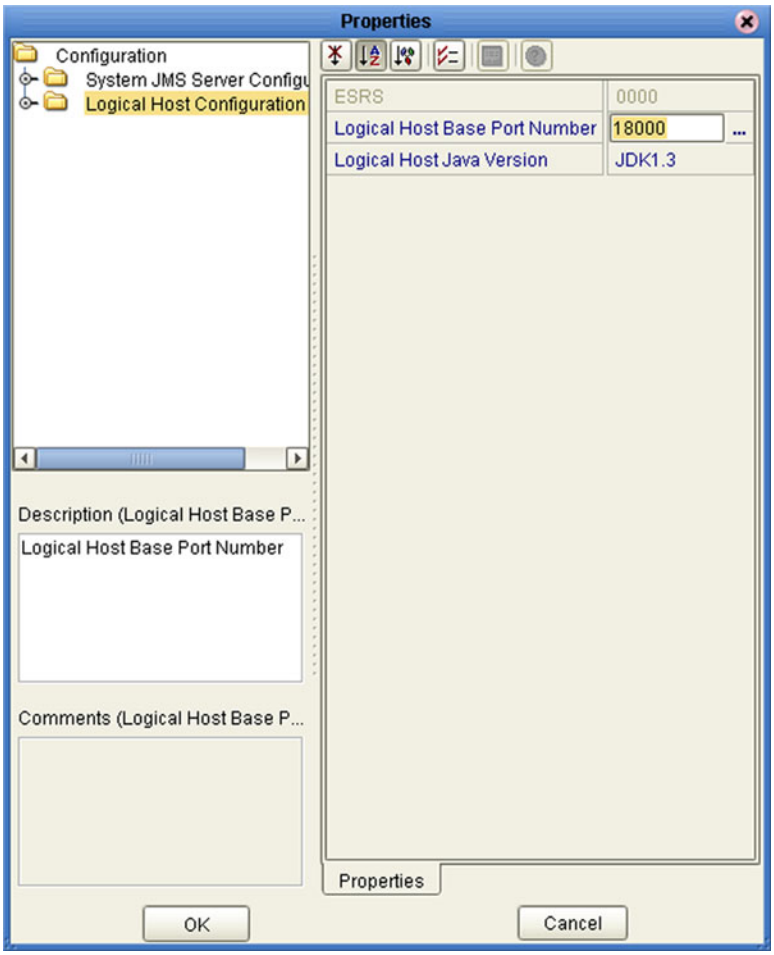
Figure 110 Uninstall as Service Script



8.6.3 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 111).

Figure 111 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, the automatic numbering process will skip the component port number you have assigned manually (*be sure this port number is not used elsewhere*).

8.7 Starting the Logical Host

The Logical Host is started by means of a bootstrap script that is specific to the platform on which it is running. You can start the Logical Host manually by opening a command prompt and typing a startup command having a string of arguments appended to it. These include both optional and required initial (first-time only) arguments, as listed in Table 28. If you have configured the logical-host.properties file (see **Modifying the Logical Host Properties File** on page 138), you can use the default values by omitting the arguments.

The startup command string uses the following syntax:

```
bootstrap argument1 ... argumentN
```

For example, the command

```
bootstrap -h
```

displays the online *Help* entry that explains the command line arguments.

Table 28 Startup Command Arguments

Parameter	Description	Req/Opt
-d *debug*	Overrides bootstrap sequence. Displays all cached (default) argument values.	Optional
-e *environment name*	The name of the Environment to which this Logical Host belongs.	Required (first time only)
-h *help*	Overrides bootstrap sequence. Displays the usage report.	Optional
-i *id*	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 *logicalhost name*	The name of this Logical Host.	Required (first time only)
-n *physical host name*	The name of this Physical Host.	Required (first time only)
-p *password*	The password used for accessing the Repository.	Required (first time only)
-r *repository URL*	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.*

8.7.1 Starting the Logical Host on a Windows System

To start the Logical Host for the first time, or to override existing default values

1. Open a command prompt on your desktop.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root\logicalhost\bootstrap\bin\
   ```

3. Type the startup command including all of the required parameters shown in Table 28:

   ```
   bootstrap argument1 ... argumentN
   ```

4. Click **Enter**.

To start the Logical Host after the first time, or to use existing default values

1. Open a command prompt on your desktop.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root\logicalhost\bootstrap\bin
   ```

3. Double-click **bootstrap.bat**.

8.7.2 Starting the Logical Host on a UNIX System

To start the Logical Host

1. Open a command prompt on your desktop.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root/logicalhost/bootstrap/bin/
   ```

3. Run the bootstrap script using the following command:

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

8.7.3 Starting the Logical Host on a Red Hat Linux System

To start the Logical Host

1. Open a command prompt on your desktop.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root/logicalhost/bootstrap/bin/
   ```

3. Run the bootstrap script using the following command:

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

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

8.8 Stopping the Logical Host

The Logical Host can be shut down either from the Enterprise Monitor or by means of the following command-line procedure.

To stop the Logical Host

1. Open a command prompt on your desktop.
2. Change to the directory containing **bootstrap.bat**:

   ```
   cd ICAN_root/logicalhost/bootstrap/bin/
   ```

3. Run the file **shutdown.bat** (or **shutdown.sh**).

Chapter 9

Project Deployment

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

9.1 Deployment Profiles

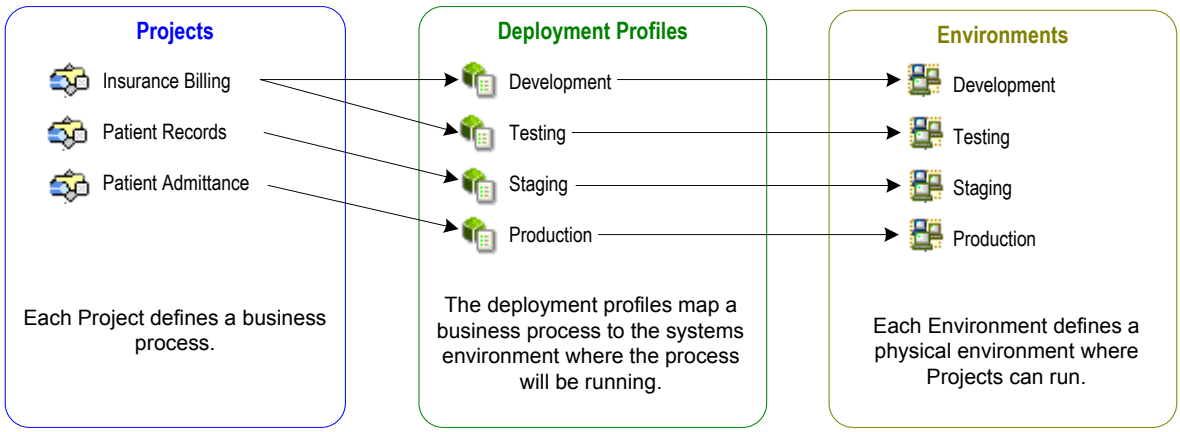
Deployment Profiles define specific instances of a Project in a particular Environment. A deployment profile contains information about the assignment of Services and Message Destinations to integration and message servers (JMS IQ Managers). It also contains version information for all relevant objects 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 19:

Figure 112 eGate Integrator Implementation Model



9.2 Using the Deployment Editor

The Deployment Editor (see Figure 113) appears when you create a new Deployment Profile or click on an existing Deployment Profile icon in the Project Explorer tab.

Figure 113 Deployment Editor Window

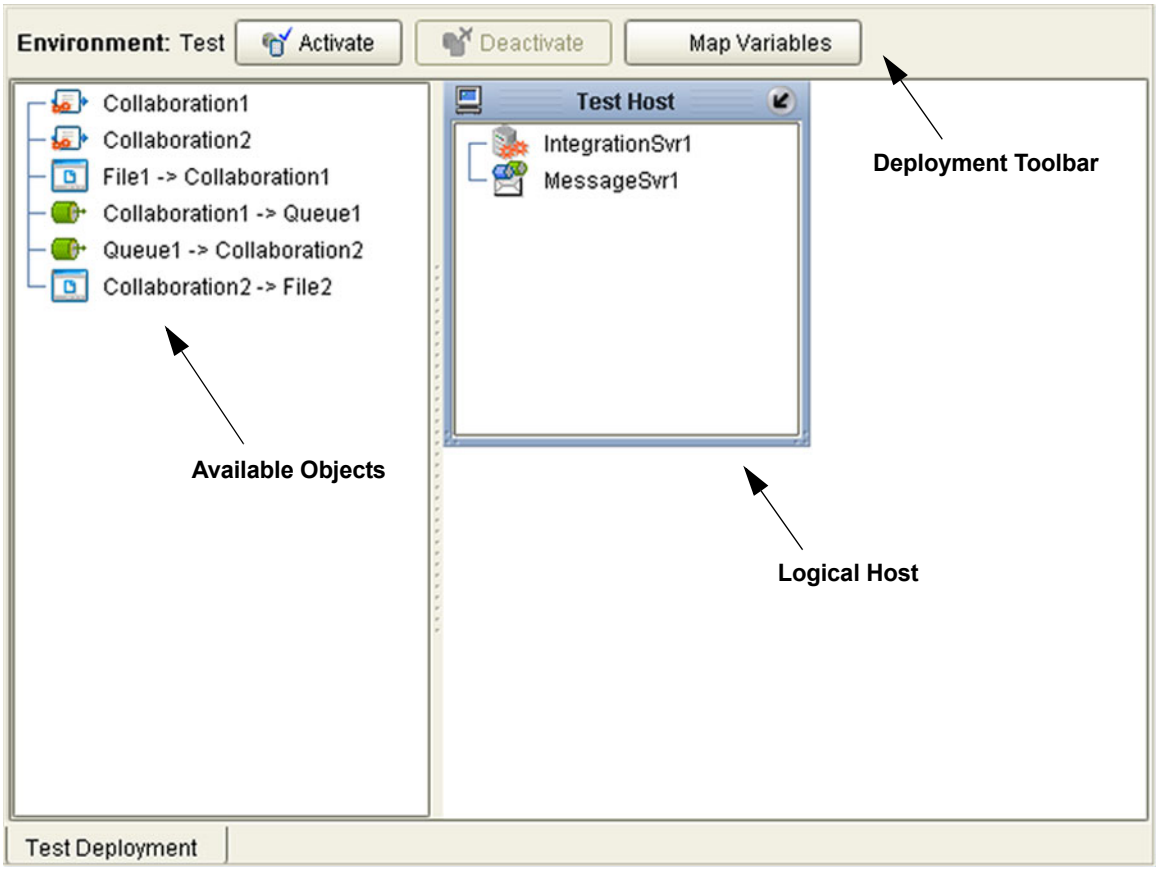


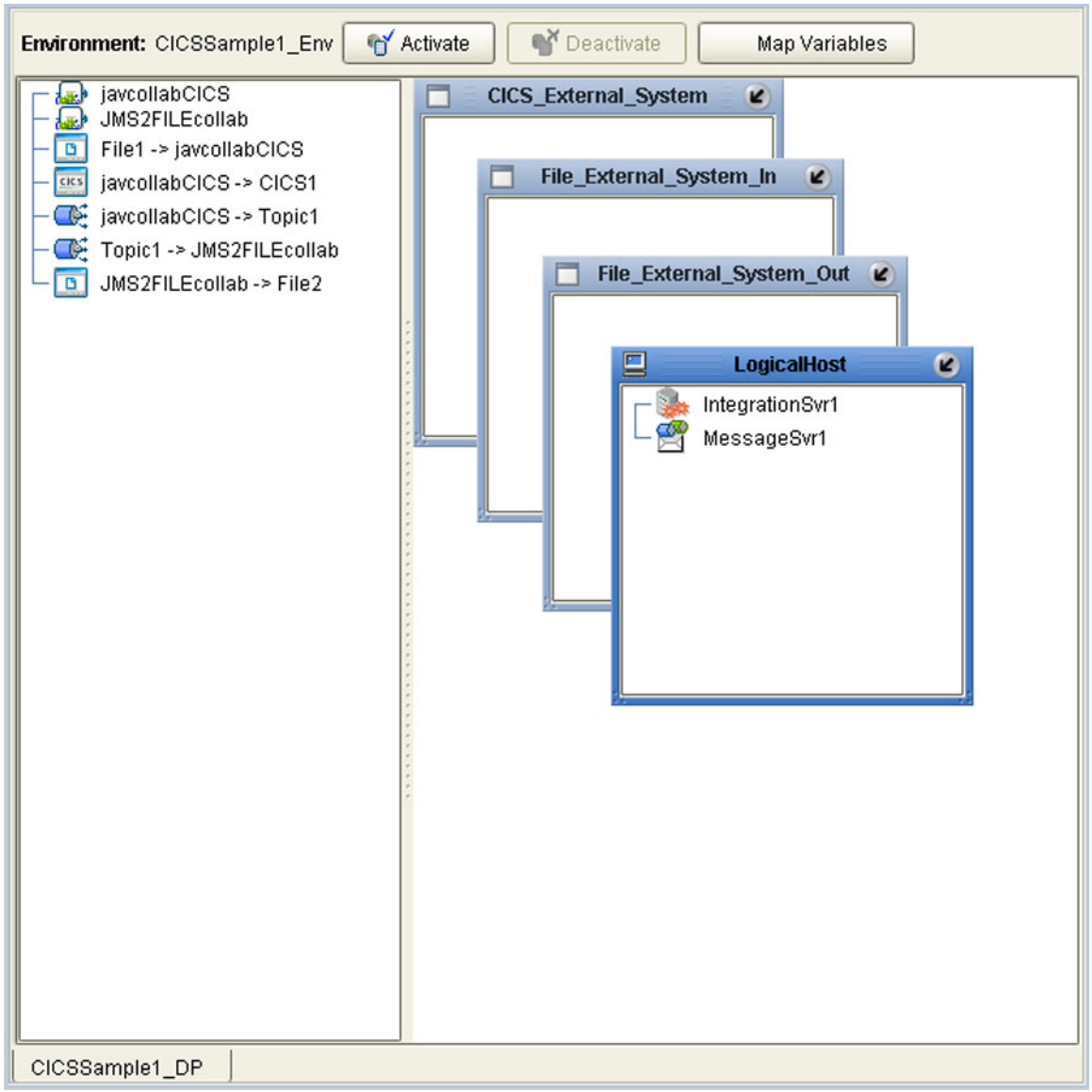
Table 29 Deployment Toolbar Buttons

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 Deployment Profiles** on page 152.
Deactivate	Stops the Project by terminating the link between the EAR file and the SeeBeyond Integration Server, sets the Deployment Profile to *inactive*, 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 154.

9.3 Creating a Deployment Profile

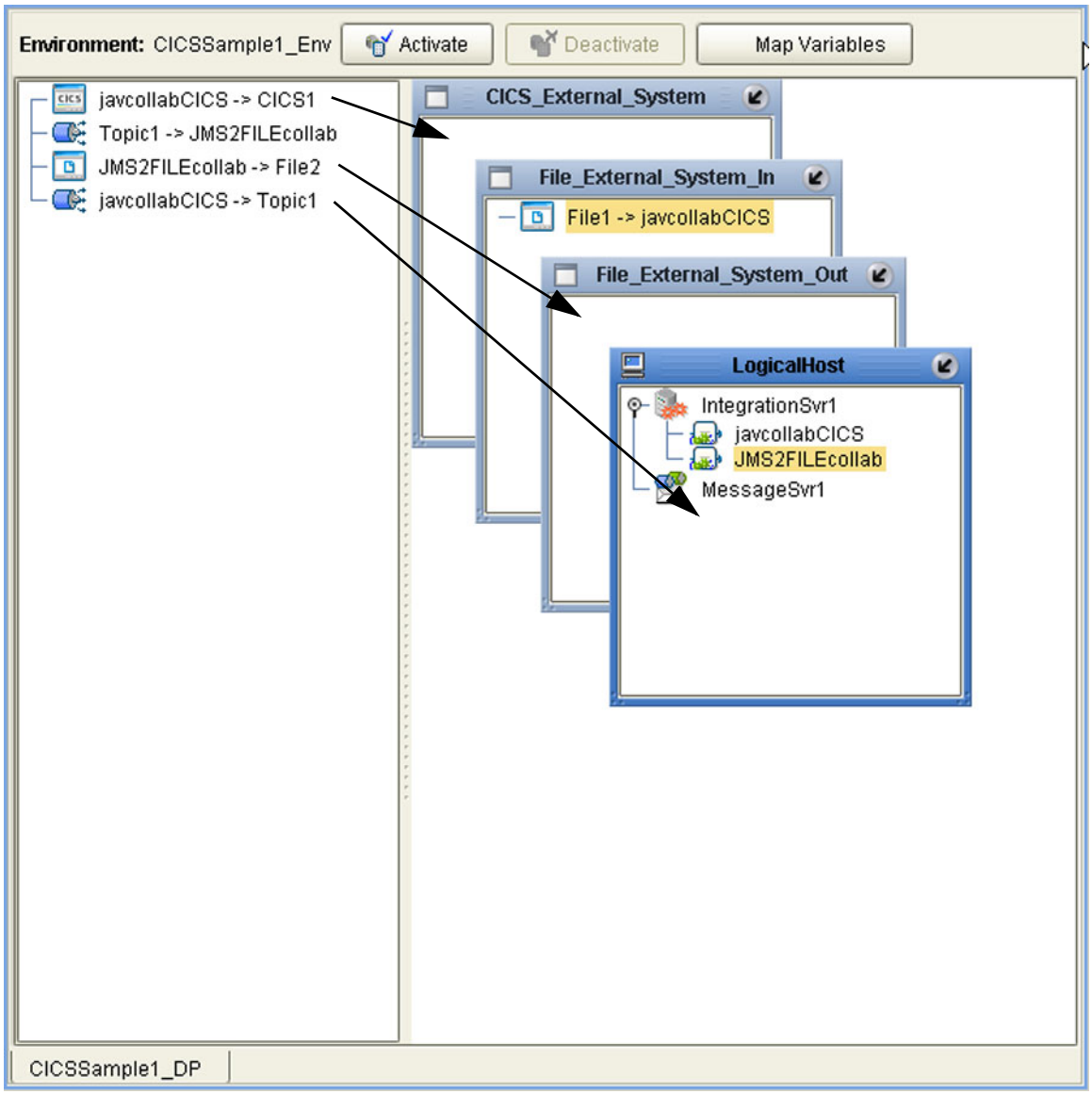
In the Environment Explorer, create an Environment and right-click on the Environment to display its context menu. From the menu, select the Environment components you need and name them appropriately. They will appear on the Deployment Editor canvas, as shown in Figure 114.

Figure 114 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 115. As you do so, they will disappear from the left panel.

Figure 115 Example Deployment Profile (2)

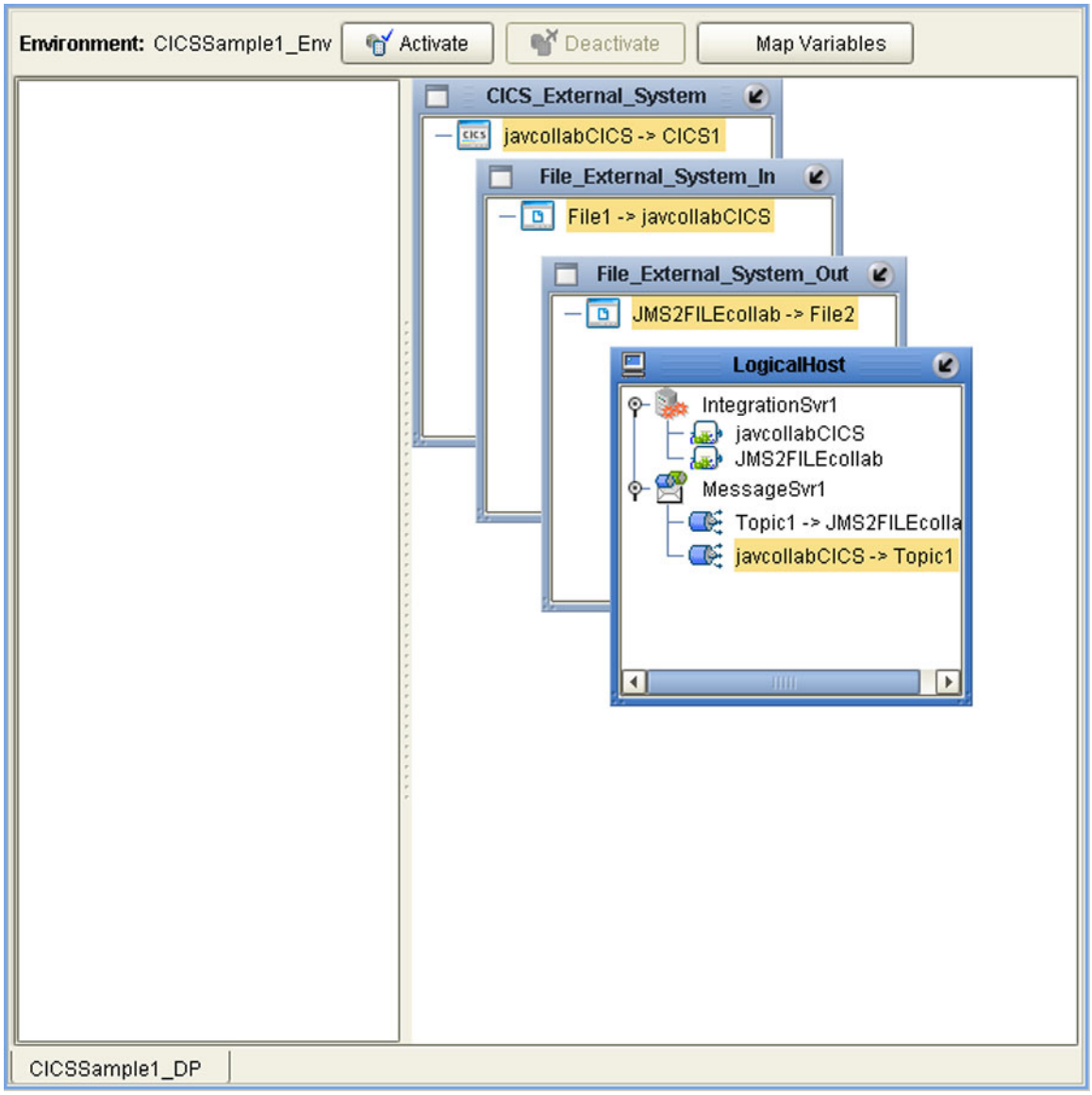


Note that:

- 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 116. The Deployment Profile is now ready to be saved and Activated.

Figure 116 Example Deployment Profile (3)



9.4 Activating and Deactivating Deployment Profiles

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.

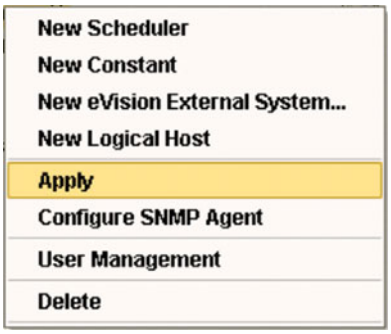
To activate a Deployment Profile

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

Figure 117 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 118). This will apply all of the changes for that Logical Host.

Figure 118 Logical Host Context Menu - Apply

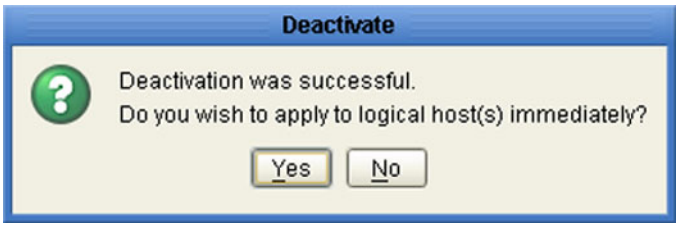


Note: *The **Apply** action assumes that the Logic Host is running, since it invokes a trigger to the Logical Host causing it to download the latest settings from the Repository and deploy those settings to all components on the Logical Host.*

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 119 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 118). 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 s 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.*

9.5 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 120) and values (see Figure 121).

Figure 120 Deployment Profile Mappings

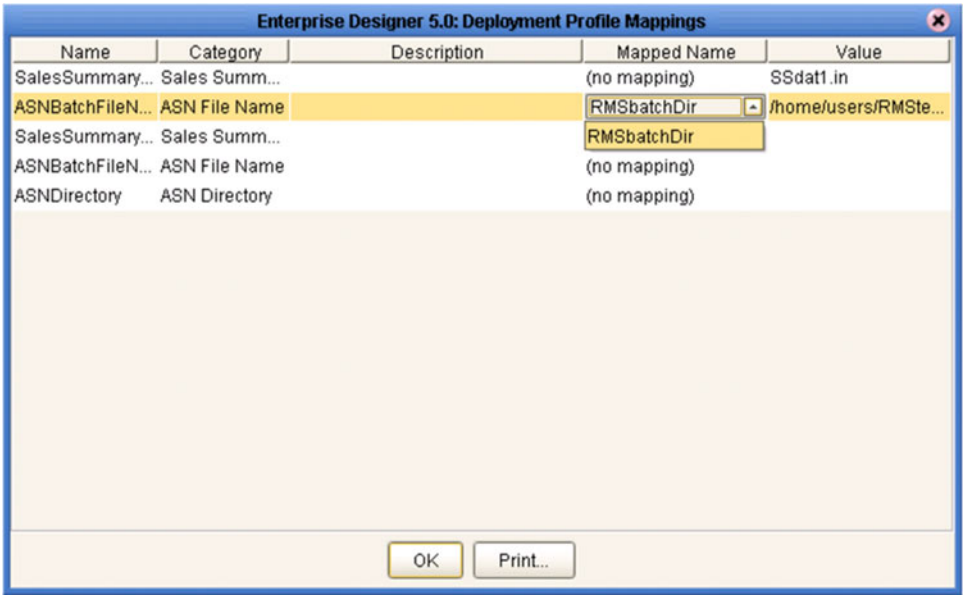
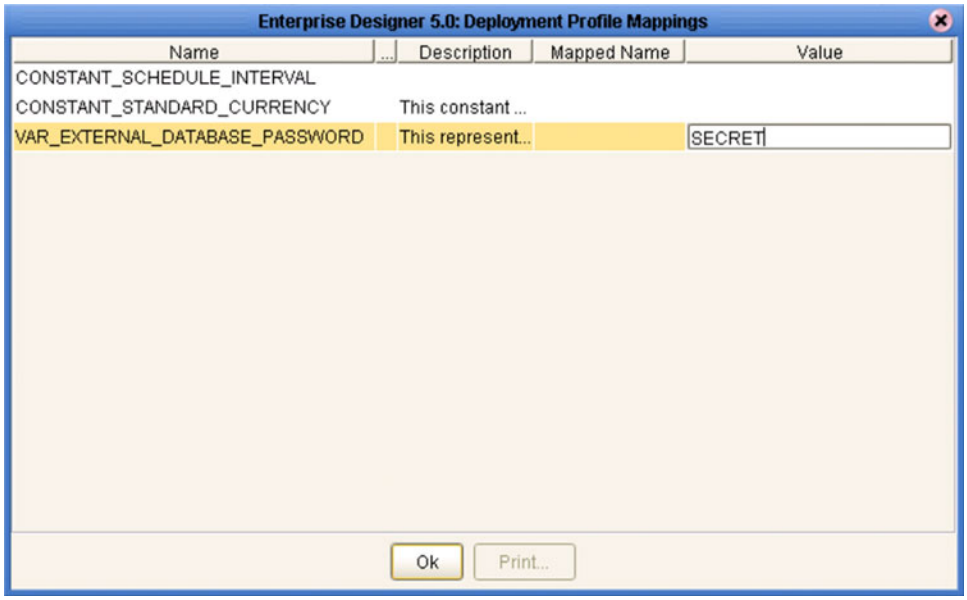


Figure 121 Project Variable Value Entry



9.6 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.*

9.6.1 BEA WebLogic

To deploy an eGate Project to a BEA WebLogic environment

1 Create the following components in Enterprise Designer (see Figure 122):

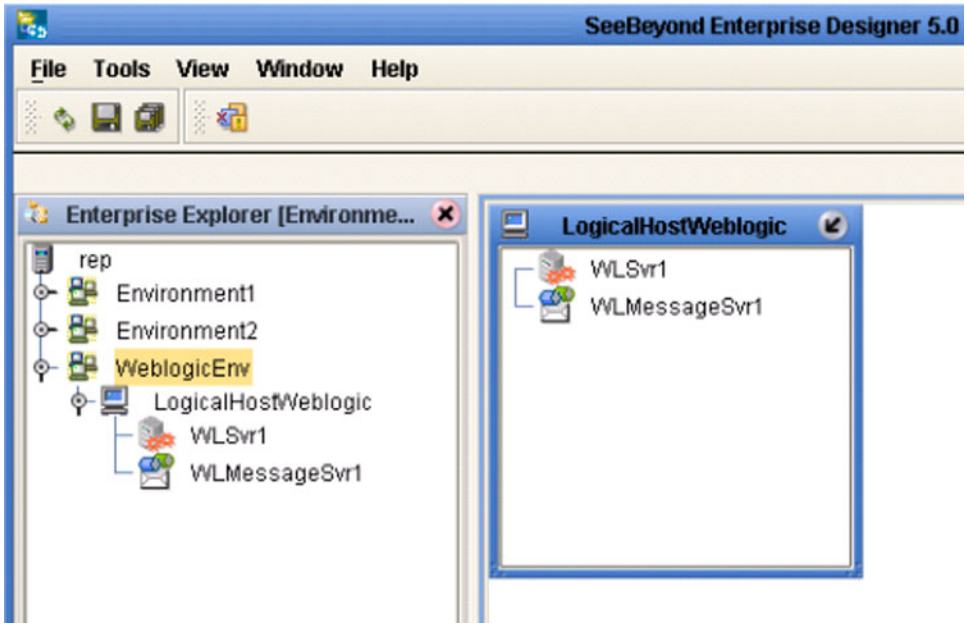
A A new environment

B A Logical Host

C A WebLogic Integration Server

D A WebLogic JMS Message Server

Figure 122 WebLogic Deployment (1)

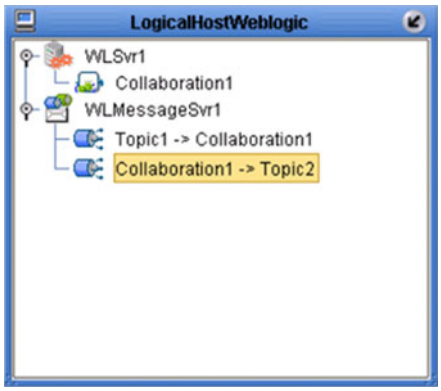


2 Create a new Deployment Profile to bind the Connectivity Map to the new WebLogic environment (see Figure 123).

A Drag the two topics and drop onto the WebLogic message server.

B Drag the Collaboration and drop onto the WebLogic integration server.

Figure 123 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:

```
ICAN-root\repository\data\files\WLEnvironmrntName\
    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.

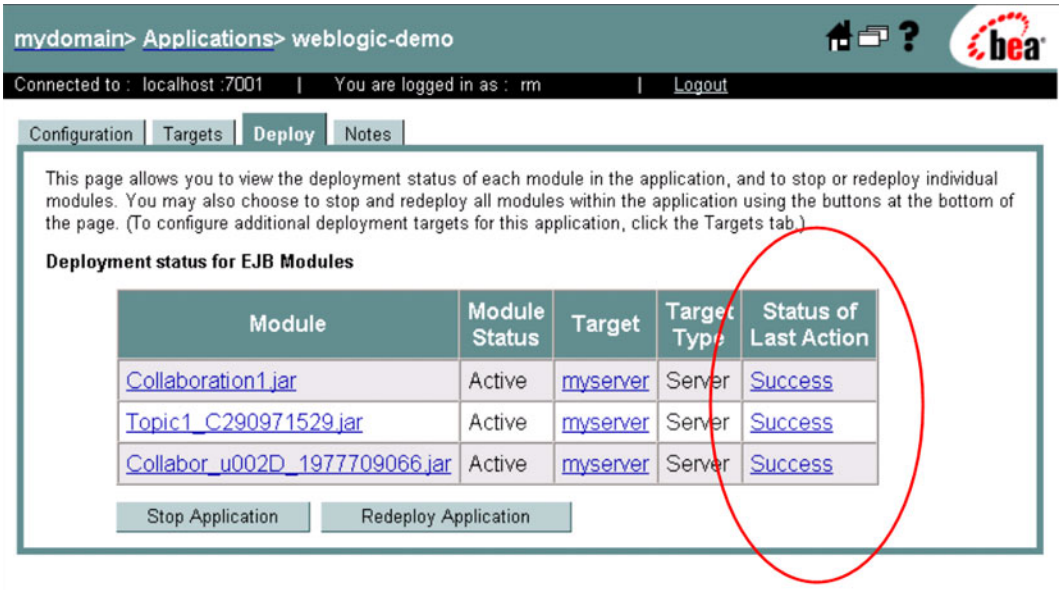
F Select the EAR file you just installed as the archive for the new application.

G Enter a name for the new application.

H Click **Deploy**.

I Verify the success of the deployment (see Figure 124 , which shows a WebLogic 8.1 example).

Figure 124 WebLogic Deployment Verification



9.6.2 IBM WebSphere

To deploy an eGate Project to an IBM WebSphere environment

1 Create the following components in Enterprise Designer (see Figure 125):

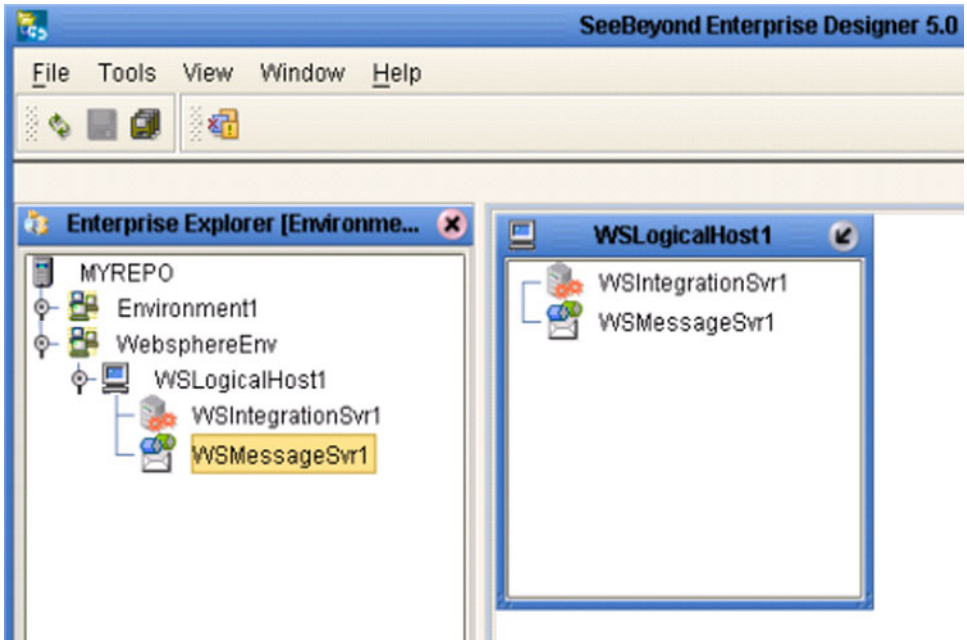
A A new environment

B A Logical Host

C A WebSphere Integration Server

D A WebSphere JMS Message Server

Figure 125 WebSphere Deployment (1)

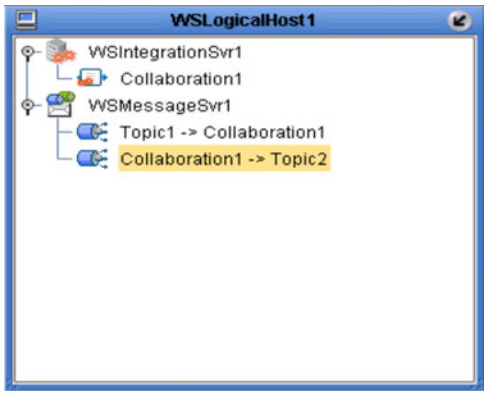


2 Create a new Deployment Profile to bind the Connectivity Map to the new WebSphere environment (see Figure 126).

A Drag the two topics and drop onto the WebSphere message server.

B Drag the Collaboration and drop onto the WebSphere integration server.

Figure 126 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:

```
ICAN-root\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_WSMessageSvr1
```

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.*

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**.

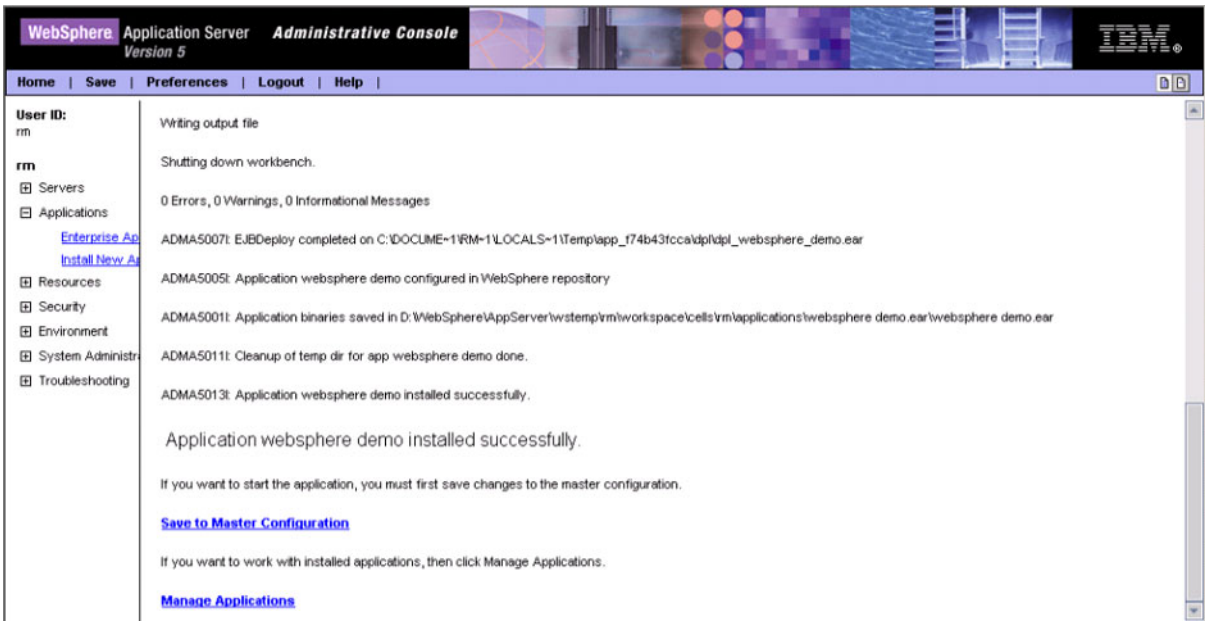
16 In *Step 7, Map modules ...*, check all modules and click **Next**.

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 127 , which shows a WebSphere 5 example).

Figure 127 WebSphere Deployment Verification



Chapter 10

ICAN Security Features

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

10.1 Overview

ICAN users can be classified into two categories:

1. Users of the ICAN Suite toolset.

   This category includes those who perform the development, administration, and management activities. The deployment and bootstrap tasks also fall into this category. These users are also logically mapped to the *all*, *administration*, and *management* roles, respectively.

2. Users of J2EE applications running in the Environment.

   This category includes those who access the deployed J2EE applications in the Logical Host in an Environment. Potentially, these users are the Customers of the Enterprise accessing the J2EE applications.

Security features described in this chapter are as follows:

- **Configuration User Management** on page 163 describes the management of users in the ICAN Suite.
- **Environment User Management** on page 188 describes the management of users who would access the applications deployed in an enterprise, using the ICAN Suite.
- **ACL Management** on page 190 describes the management of access control to various components and features in the ICAN Suite.
- **ICAN Component Security** on page 193 briefly describes the security settings for message servers and JMS Client connections. Additional information is contained in the *eGate Integrator JMS Reference Guide*.
- **Using SSL/HTTPS in ICAN** on page 195 describes the use of Secure Sockets Layer protocols in Web communications.

10.1.1 Multiple Environments

Deploying Projects to multiple Environments requires special considerations regarding security.

To prepare for deployment to multiple Environments

1. Create the users who will develop, administer, or manage the multiple Environments in the Repository.
2. Set the ACL on the Environments to isolate them and grant access to only the specific Environment users (such as Administrators).
3. Create the J2EE application-specific users and roles in the respective Environments.
4. Set the environment-specific settings for the application using the users and roles you have created for the Environment.

10.2 User Management

10.2.1 Configuration User Management

In order to access the ICAN Suite, an individual must be registered as a *user* in the ICAN security system by a system administrator. Once entered into the system, the user can then be assigned privileges allowing access to different parts and features of the ICAN Suite. User management takes effect immediately, so it is not necessary to reboot the Repository to reflect any changes.

Roles

The Enterprise Designer contains facilities that allow a system administrator to manage user access, based on *roles* and user IDs. There are three predefined roles in the ICAN Suite, as described in the following table. Note that a user can have more than one role, in which case his privileges will be the combined privileges from both roles.

Role	Description
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. This is the most basic role, and offers the minimum permission level.
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.

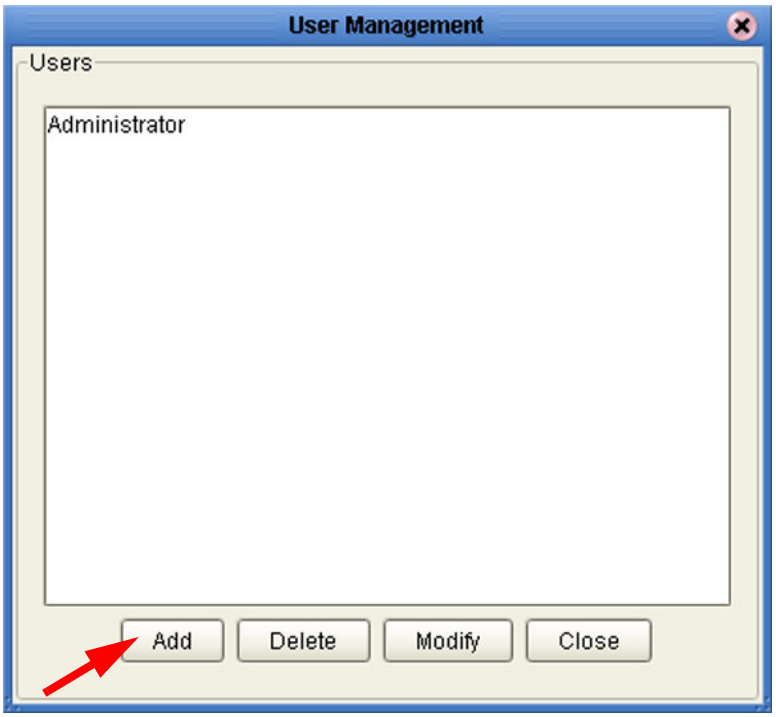
A mechanism is included in the Enterprise Designer for ICAN 5.0 to create roles in addition to the three predefined roles described above. The role you create using this mechanism provides a means for organizing users into groups that can be referenced in your own Web-based application (or perhaps in a third-party application that you are using). Future versions of the ICAN Suite will make additional use of this capability within the Suite itself. See **Creating User Roles** on page 172.

Adding and Deleting Repository Users

To add a new user

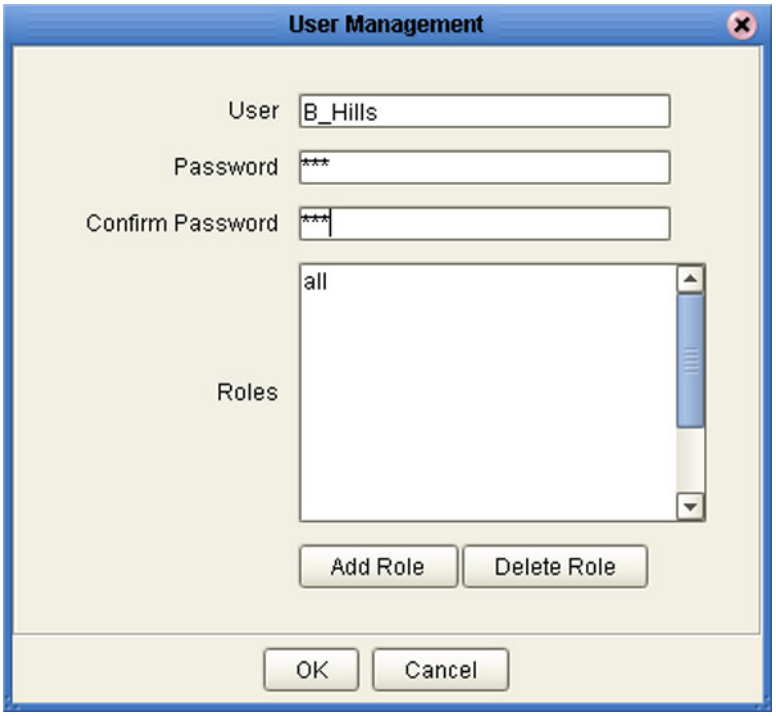
1. In the Project Explorer, select the **Repository** icon.
2. Right-click to display the Repository context menu.
3. Select **User Management** to display the User Management dialog box shown in Figure 128.

Figure 128 User Management Dialog Box (1)



1 Click **Add** to display a second User Management dialog box shown in Figure 129.

Figure 129 User Management - Add User



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.

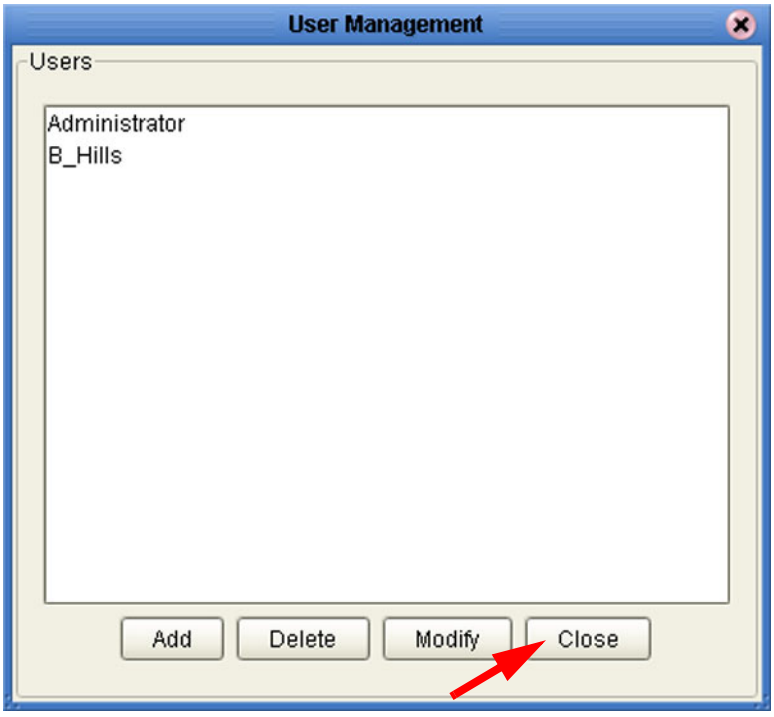
3 Enter a code for the user in the **Password** box. This is the code the user will enter as their password during system login.

Note: *Every user entered into the system is automatically assigned to the* ***all*** *role, which is required to connect to the Repository.*

4 Click **OK** to create the new user. This user can now access the Enterprise Designer and Repository with the assigned login ID and password, and this user name will be added to the list in the initial User Management dialog box (see Figure 130).

5 To add an another role for this user, see **Adding and Deleting User Roles** on page 167.

Figure 130 User Management Dialog Box (2)



6 Click **Close** to return to the Enterprise Designer.

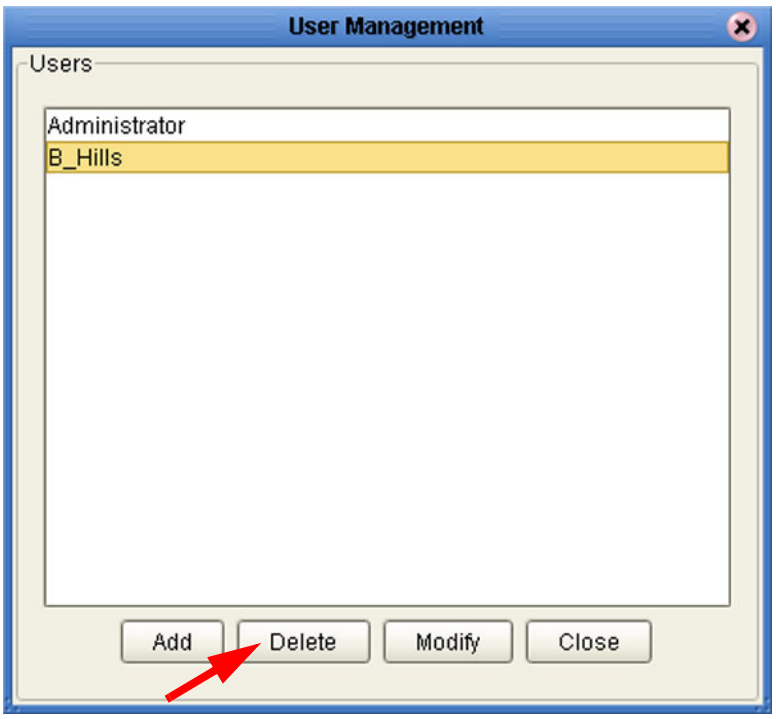
To delete a current user

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 131.

Figure 131 User Management Dialog Box

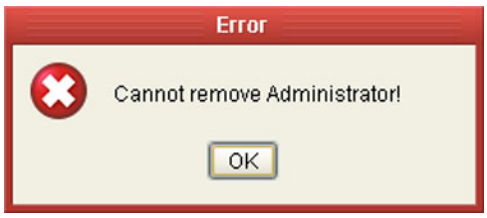


4 To delete a current user, select the user from the displayed list and click **Delete**. The user entry will be removed from the list.

5 Click **Close** to return to the Enterprise Designer.

Note: *The Administrator cannot be deleted from the system (see Figure 132).*

Figure 132 Error Warning Box



Adding and Deleting User Roles

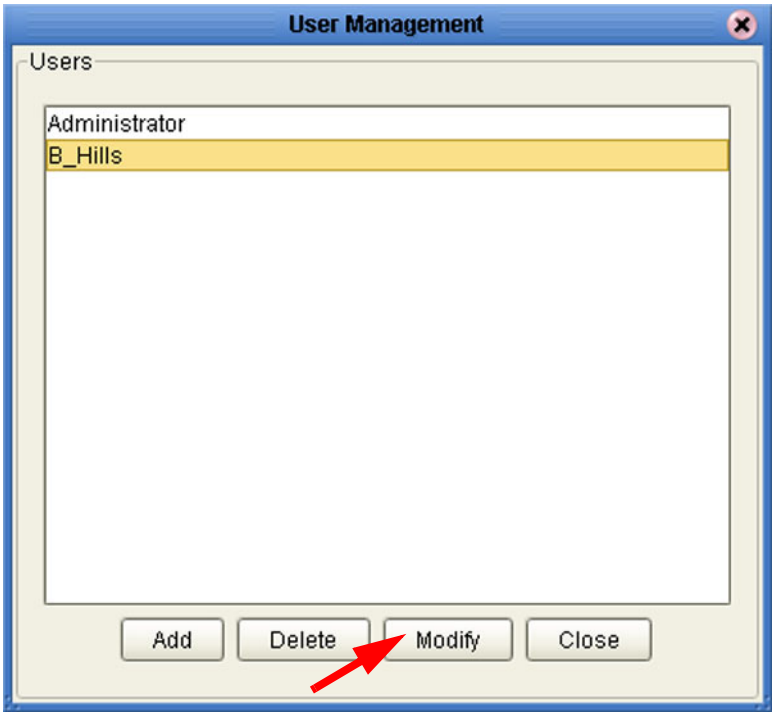
To add a new role for a current user

1 In the Project Explorer, select the **Repository** icon.

2 Right-click to display the Repository context menu.

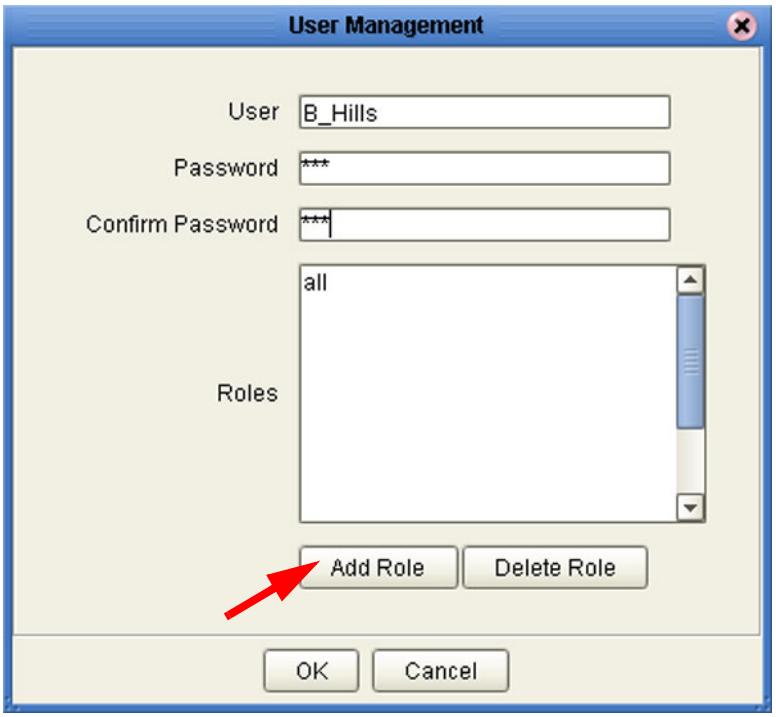
3 Select **User Management** to display the User Management dialog box shown in Figure 133.

Figure 133 User Management Dialog Box



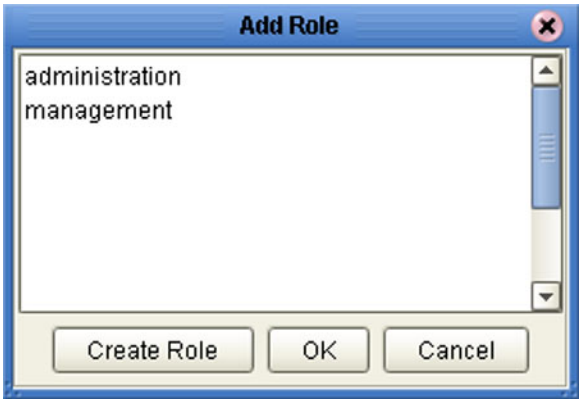
4 To add a role for a current user, select the user from the displayed list and click **Modify** to display the second User Management dialog box shown in Figure 134.

Figure 134 User Management - Add Role (1)



5 Click the **Add Role** button, which displays the dialog box shown in Figure 135.

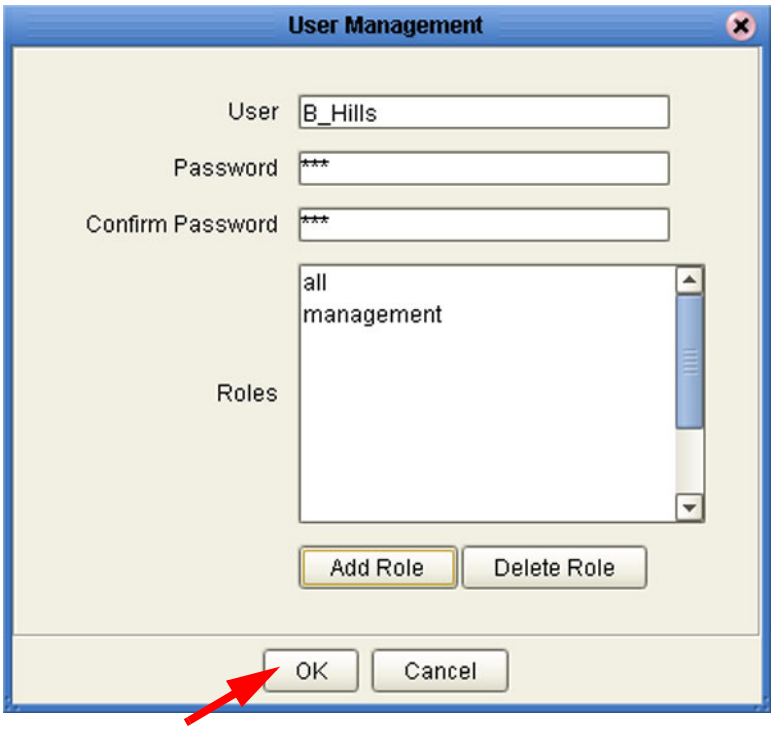
Figure 135 Add Role Dialog Box



6 Select the desired role, for example **management**, and click **OK**. The new role appears in the list for the selected user (see Figure 136).

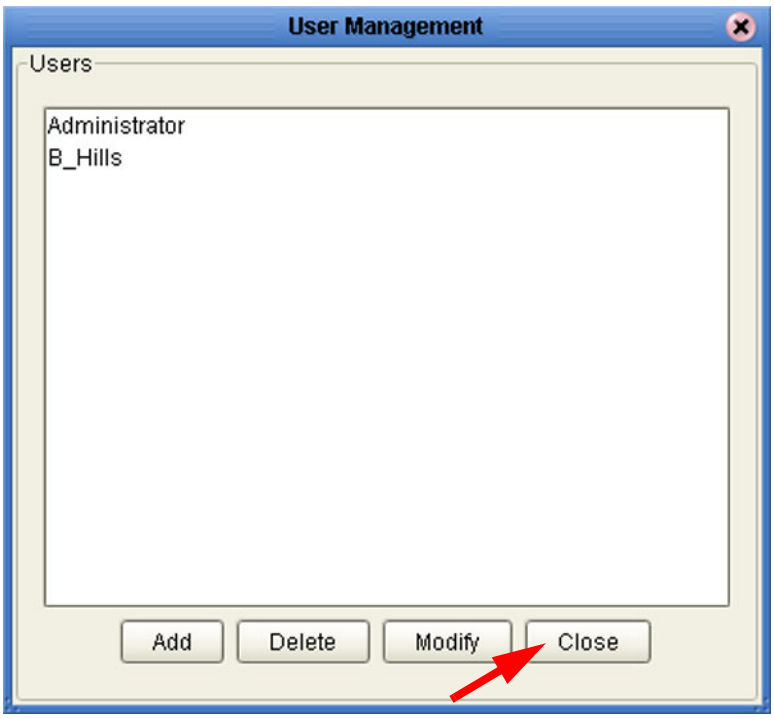
Note: *If the desired role is not listed in the Add Role dialog box, you can create a new role; see* **Creating User Roles** *on page 172.*

Figure 136 User Management - Add Role (2)



7 Click **OK** to return to the initial User Management dialog box.

Figure 137 User Management Dialog Box



8 Click **Close** to return to the Enterprise Designer.

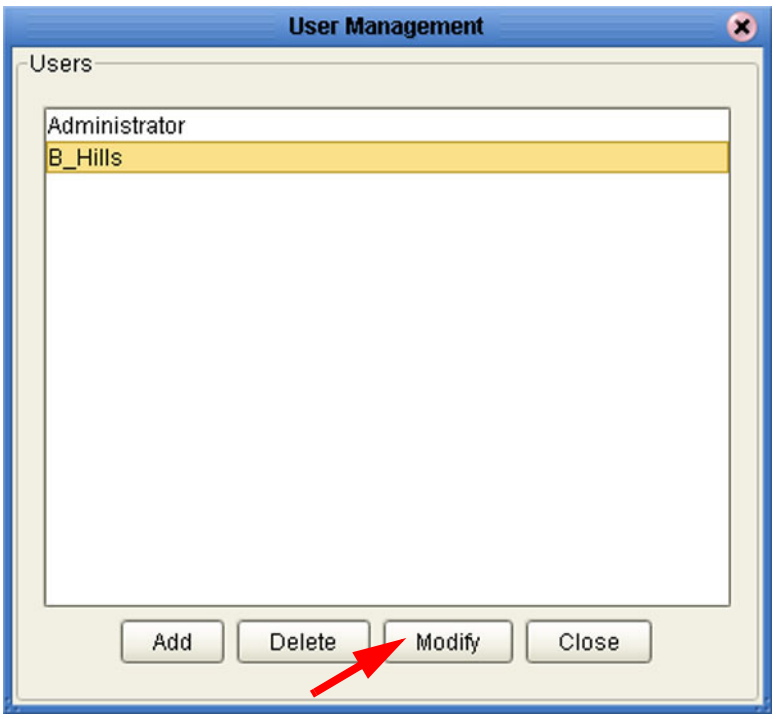
To delete an existing role for a current user

1 In the Project Explorer, select the **Repository** icon.

2 Right-click to display the Repository context menu.

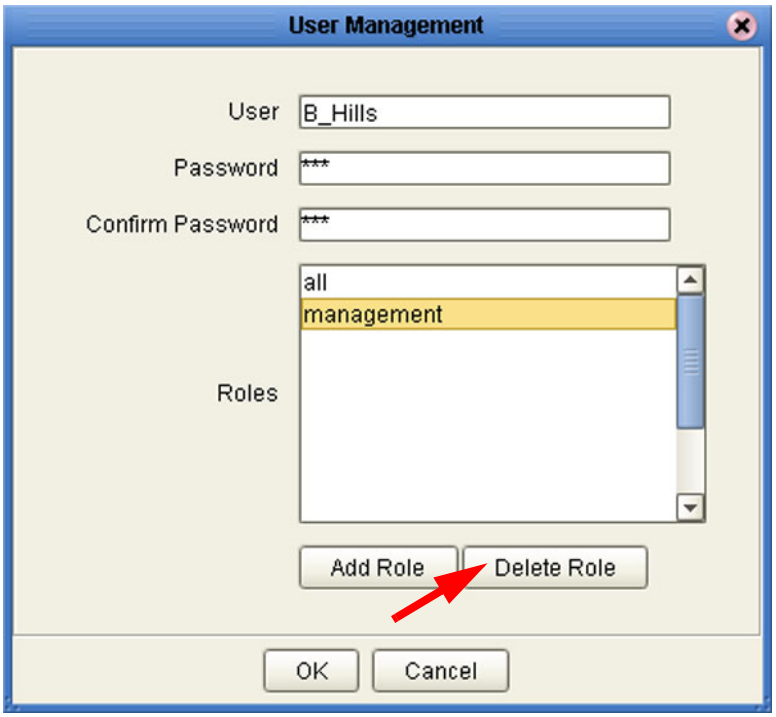
3 Select **User Management** to display the User Management dialog box shown in Figure 138.

Figure 138 User Management Dialog Box



4 To delete a role for a current user, select the user from the displayed list and click **Modify** to display the second User Management dialog box shown in Figure 139.

Figure 139 User Management - Delete Role



5 Select the role to be deleted, and click the **Delete Role** button to delete the role for the selected user. The role will disappear from the list.

6 Click **OK** to return to the initial User Management dialog box.

7 Click **Close** to return to the Enterprise Designer.

Note: *Every user entered into the system 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 (see Figure 140).*

Figure 140 Error Warning Box

Creating User Roles

New roles having limited functionality can be created for your own use. See the discussion in **Roles** on page 163.

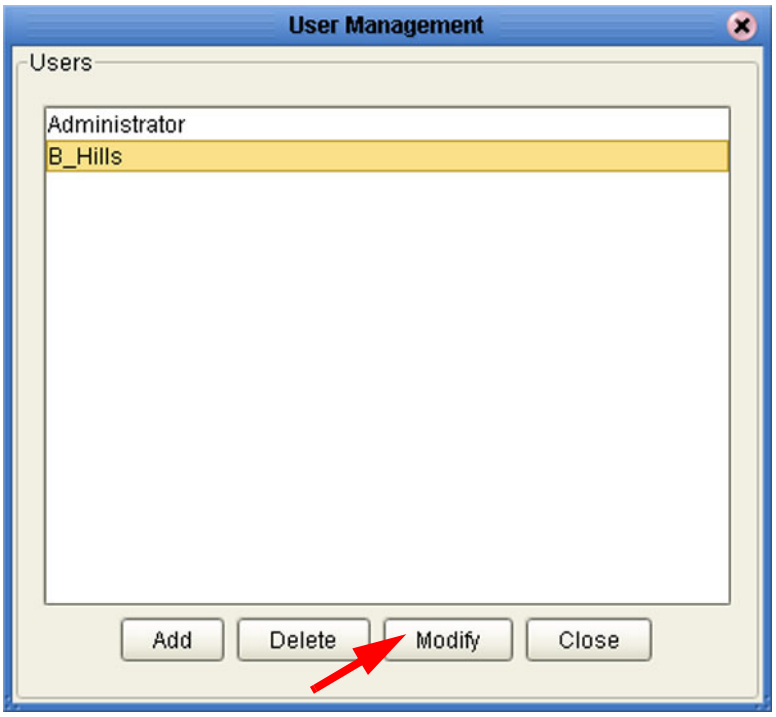
To create a new role for a current user

1 In the Project Explorer, select the **Repository** icon.

2 Right-click to display the Repository context menu.

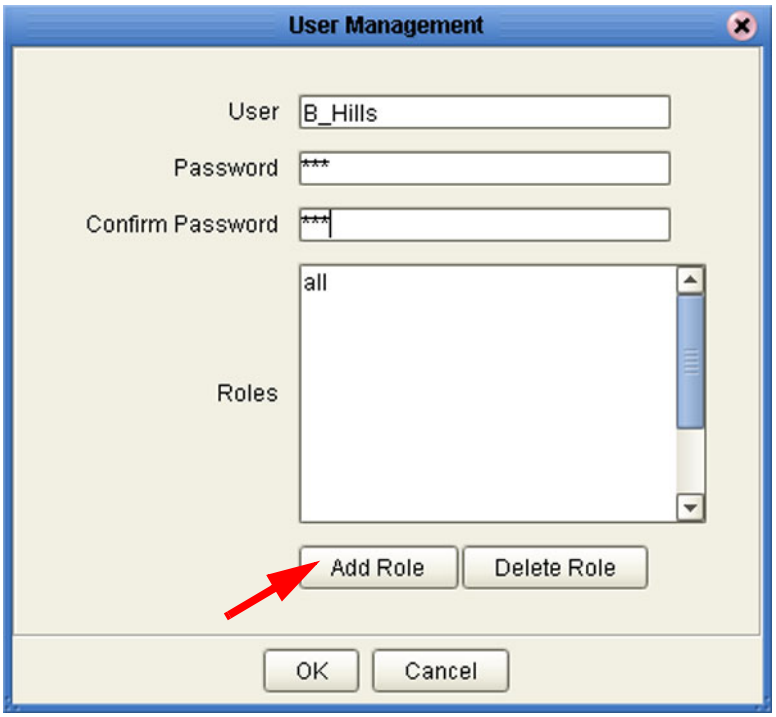
3 Select **User Management** to display the User Management dialog box shown in Figure 141.

Figure 141 User Management Users List



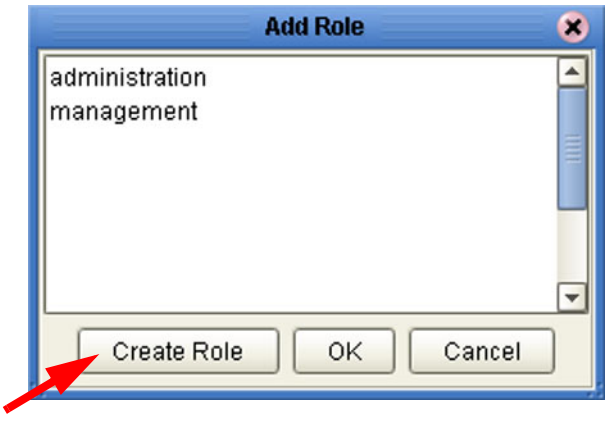
4 Select the desired user from the displayed list and click **Modify** to display the second User Management dialog box shown in Figure 142.

Figure 142 User Management - Create Role (1)



5 Click the **Add Role** button, which displays the dialog box shown in Figure 143.

Figure 143 Add Role Dialog Box (1)



6 Click the **Create Role** button, which displays the dialog box shown in Figure 144.

Figure 144 Role Dialog Box

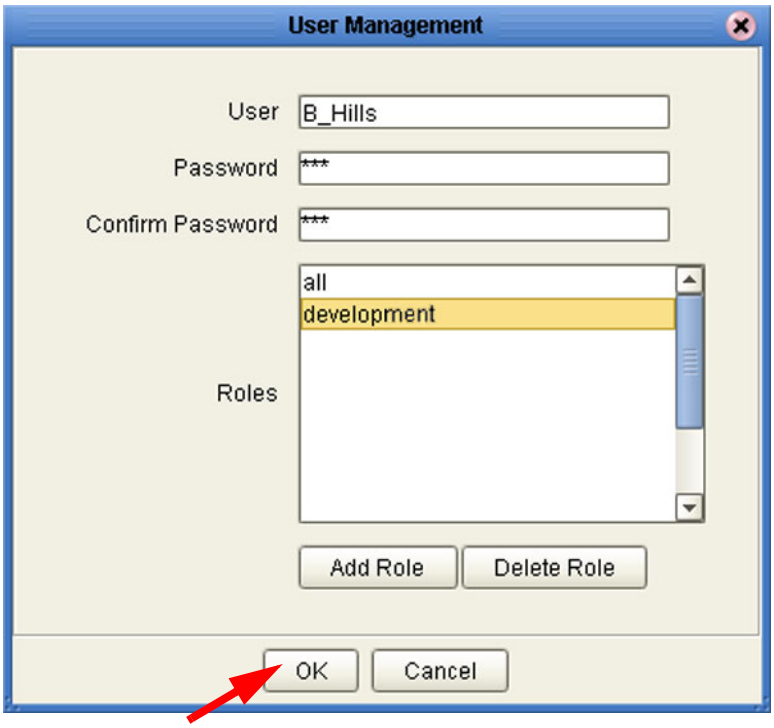
7 Type in the name of the new role you are creating, for example **development**.

8 Click **OK** to return to the *Add Role* dialog box, where the new role has been added to the list (see Figure 145).

Figure 145 Add Role Dialog Box (2)

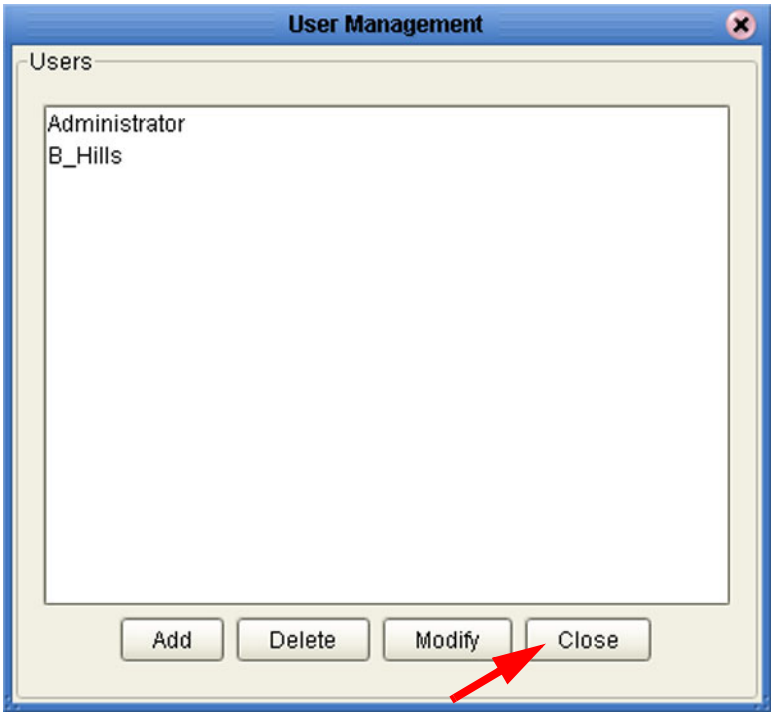
9 Select the new role and click **OK**. The role is now added for the selected user (see Figure 146).

Figure 146 User Management - Create Role (2)



10 Click **OK** to return to the initial User Management dialog box.

Figure 147 User Management Users List



11 Click **Close** to return to the Enterprise Designer.

Using LDAP

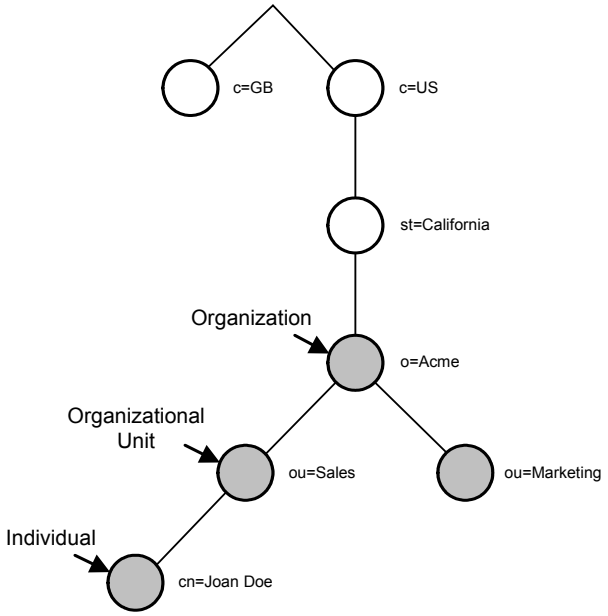
The Lightweight Directory Access Protocol (LDAP) is a protocol for accessing X.500-based directory services. LDAP runs over TCP/IP or other connection oriented transfer services. LDAP provides a mechanism for a client to authenticate, or prove, its identity to a directory server, paving the way for rich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

The LDAP 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.

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 Joan Doe. A mail attribute might contain the value jdoe@example.com. A **jpegPhoto** attribute would contain a photograph in the JPEG (binary) format.

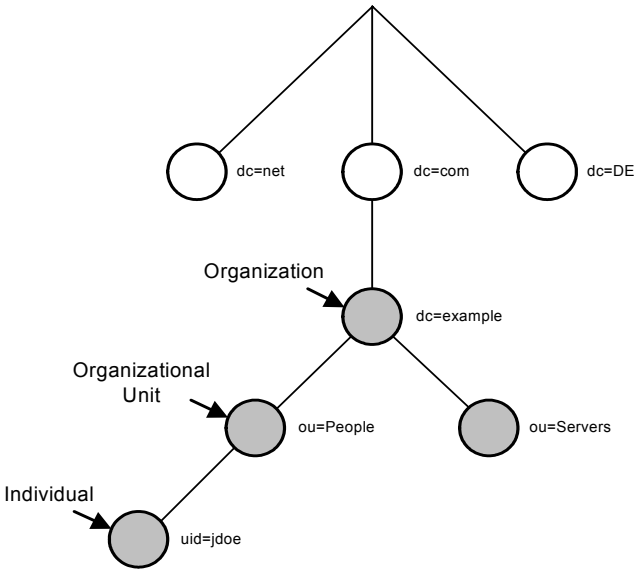
In LDAP, directory entries are arranged in a hierarchical tree-like structure. Traditionally, this structure reflected the geographic and/or organizational boundaries. Entries representing countries appear at the top of the tree. Below them are entries representing states and national organizations. Below them might be entries representing organizational units, people, printers, documents, or just about anything else you can think of. Figure 148 shows an example LDAP directory tree using traditional naming.

Figure 148 LDAP Directory Tree (Traditional Naming)



The tree may also be arranged based upon Internet domain names. This naming approach is becoming increasing popular, since it allows for directory services to be located using the DNS. Figure 149 shows an example LDAP directory tree using domain-based naming.

Figure 149 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.

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 Joan Doe in the previous Internet naming example has an RDN of uid=jdoe and a DN of uid=jdoe,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 Joan Doe, 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 do either.

Configuring LDAP Servers for Configuration User Management

ICAN 5.0 supports Microsoft's Active Directory Server (ADS) and Sun Microsystem's Sun ONE Directory Server. When a user attempts to log into the Repository, his user name and password are checked against the user name and password that is stored in either Active Directory or the Sun ONE Directory Server. The list of roles for that user is also retrieved from the respective server to authorize his access to various objects in the repository.

By default, the ICAN Suite comes with the three pre-configured roles *all*, *administration*, and *management*. Additionally, ICAN users are also allowed to create their own, custom roles. To support these roles as defined in the ICAN Suite, you must create the roles in the LDAP servers (we will demonstrate only the predefined roles).

Configuring the Active Directory Server

ADS does not support the concept of Roles, so you must simulate Roles in ADS using the ADS concept of Groups. To avoid the confusion of ADS's own Groups and ICAN's Roles, the ICAN Roles need to be located under a directory other than the ADS Groups directory.

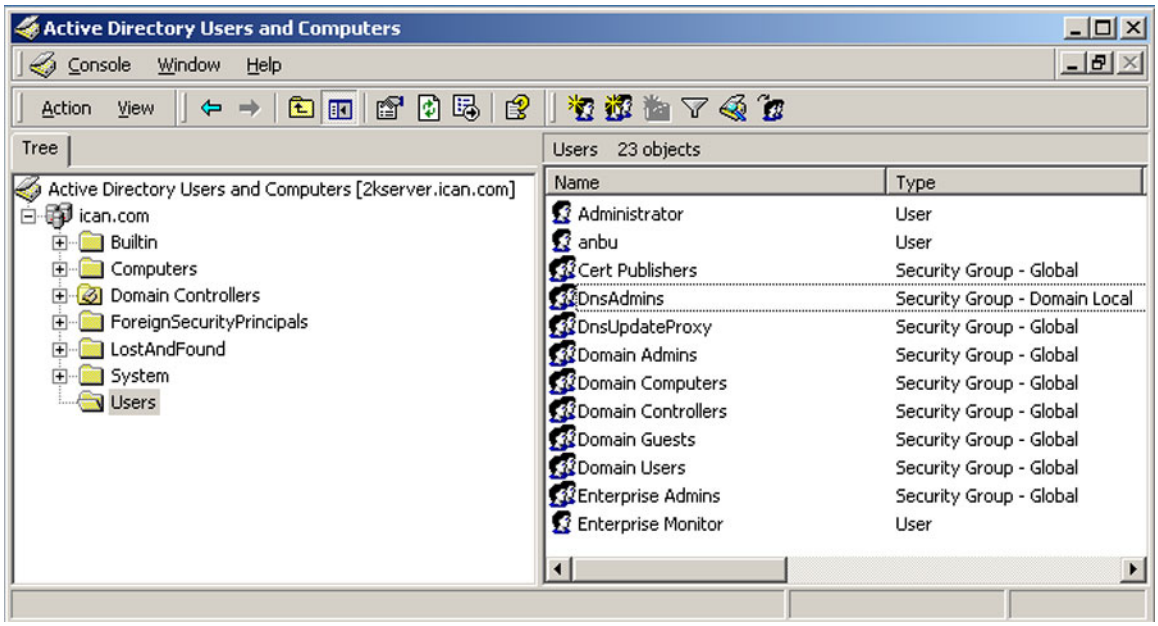
To create the ICAN Roles under their own node in ADS

1 Start the Active Directory User Management console by following the path indicated below on the computer where the Active Directory Server is running:

Start > Programs > Administrative Tools > Active Directory Users and Computers

This displays the window shown in Figure 150.

Figure 150 Active Directory Users and Computers Window

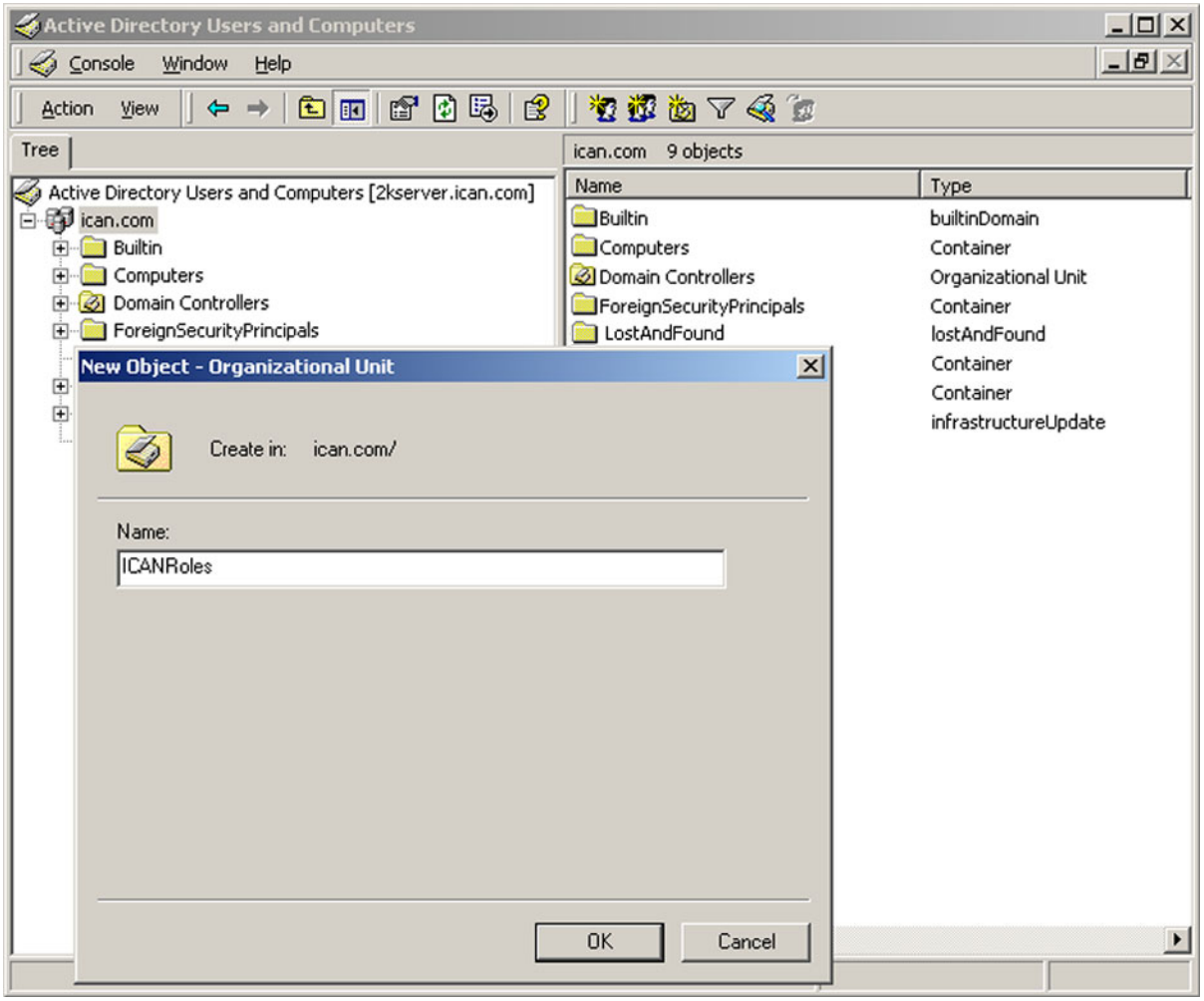


Note: *"ican.com" is a fictitious URL.*

2 On the root node, right-click and select **New > Organizational Unit** to display the *New Object - Organization Unit* dialog box (see Figure 151).

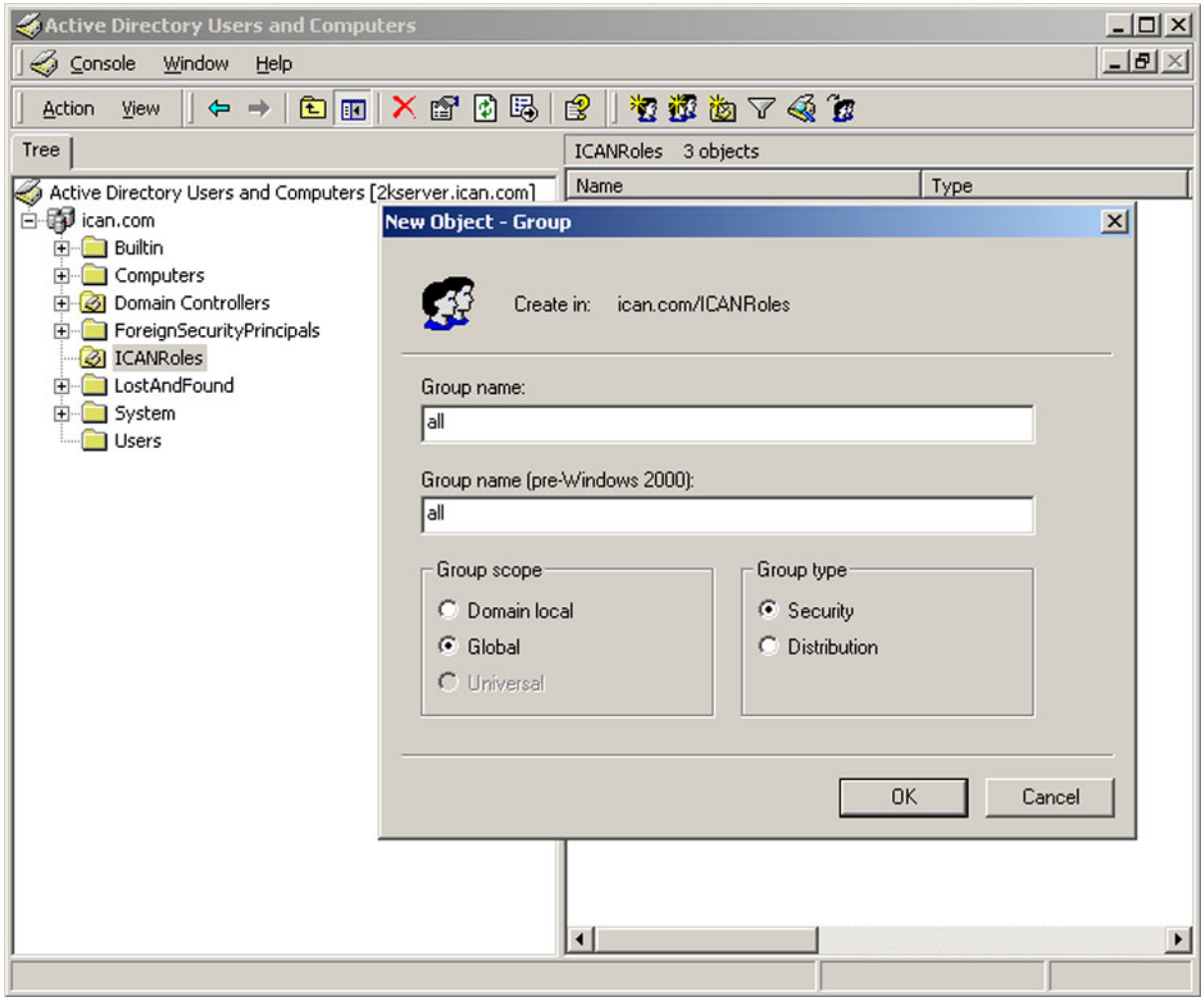
3 Enter **ICANRoles** for the Name and click **OK**.

Figure 151 Active Directory - Create Organizational Unit



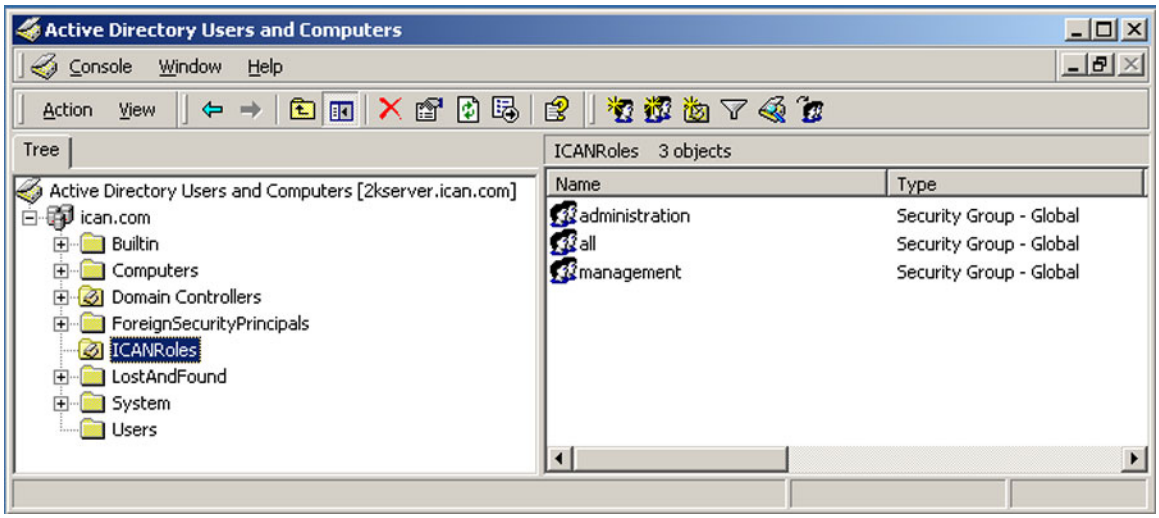
4 Under the *ICANRoles* directory, create new groups **all**, **administration**, and **management** (see Figure 152).

Figure 152 Active Directory - Create *Roles* Groups



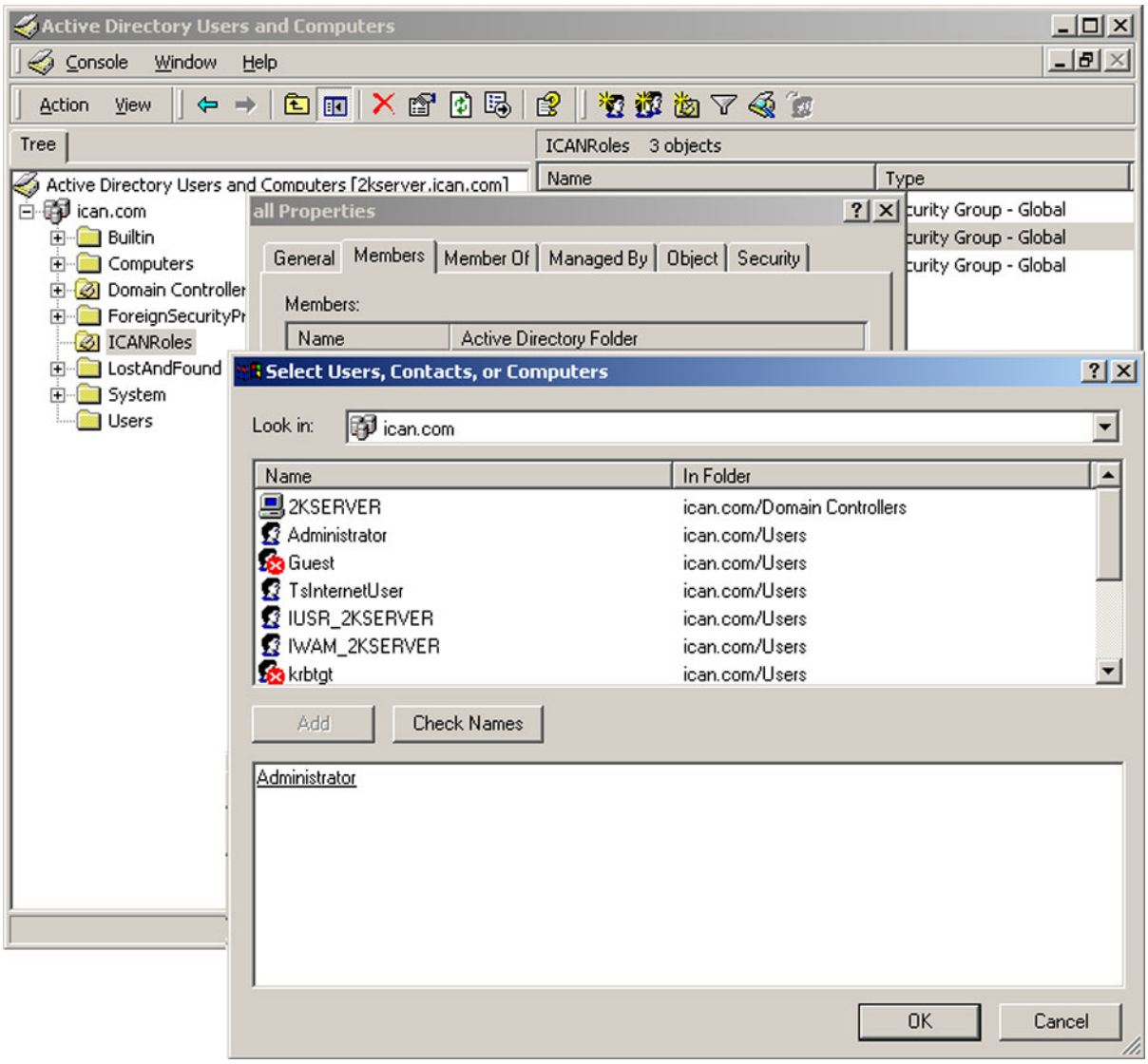
After adding all the three groups, you will see them under the *ICANRoles* as shown in Figure 153.

Figure 153 Active Directory - ICANRoles Directory



5 Add the *Administrator* user as a member of these groups by double-clicking on each of the groups and selecting **Administrator** from the dialog box (see Figure 154).

Figure 154 Active Directory - Select Administrator



6 Configure the Active Directory Server for *anonymous read*.

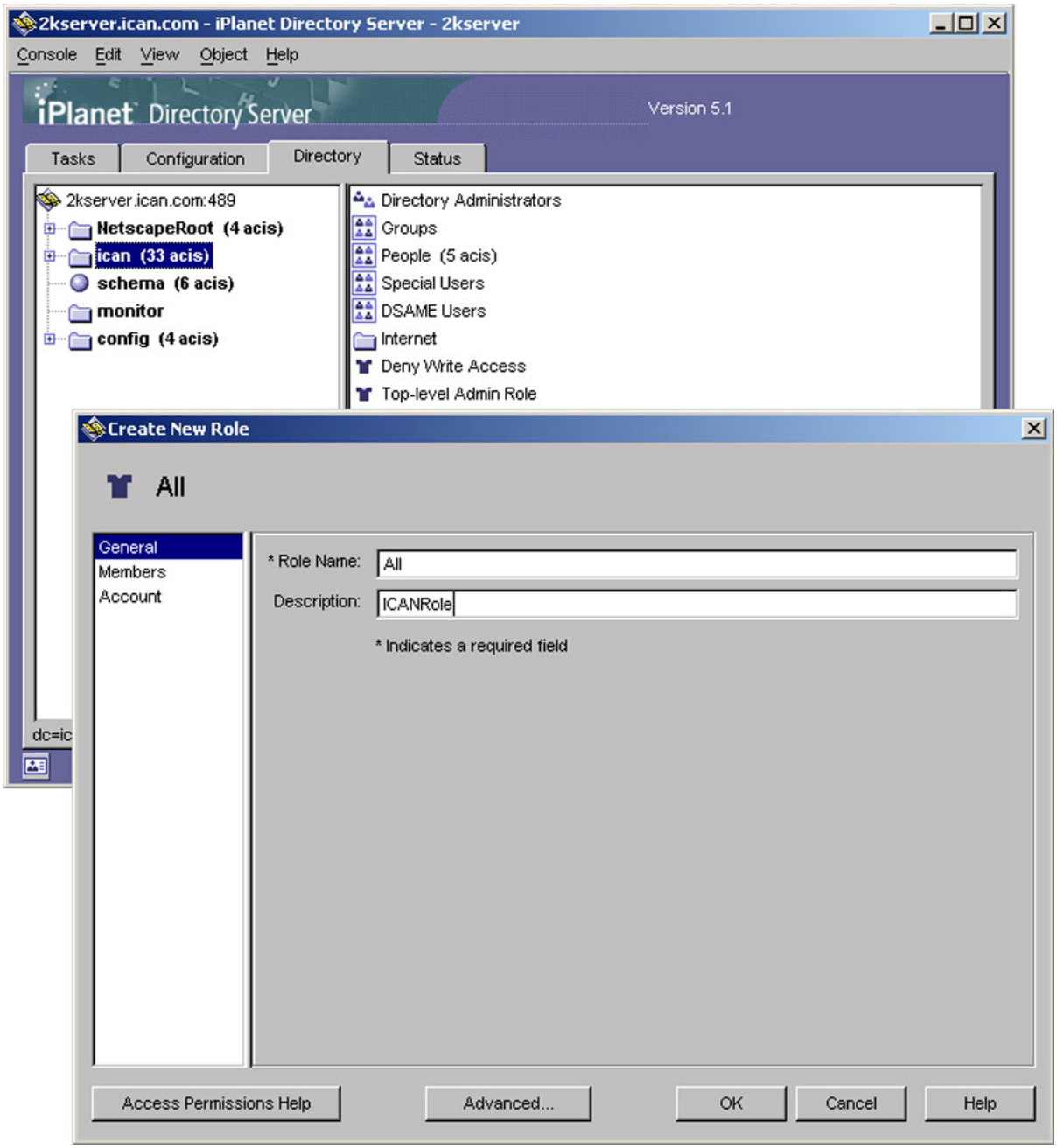
Configuring the SunONE Server

To create the ICAN Roles in the SunONE Directory Server

1 Create the user **Administrator** under the *People* directory.

2 Create the roles **all**, **administration**, and **management** under the top node as shown in Figure 155.

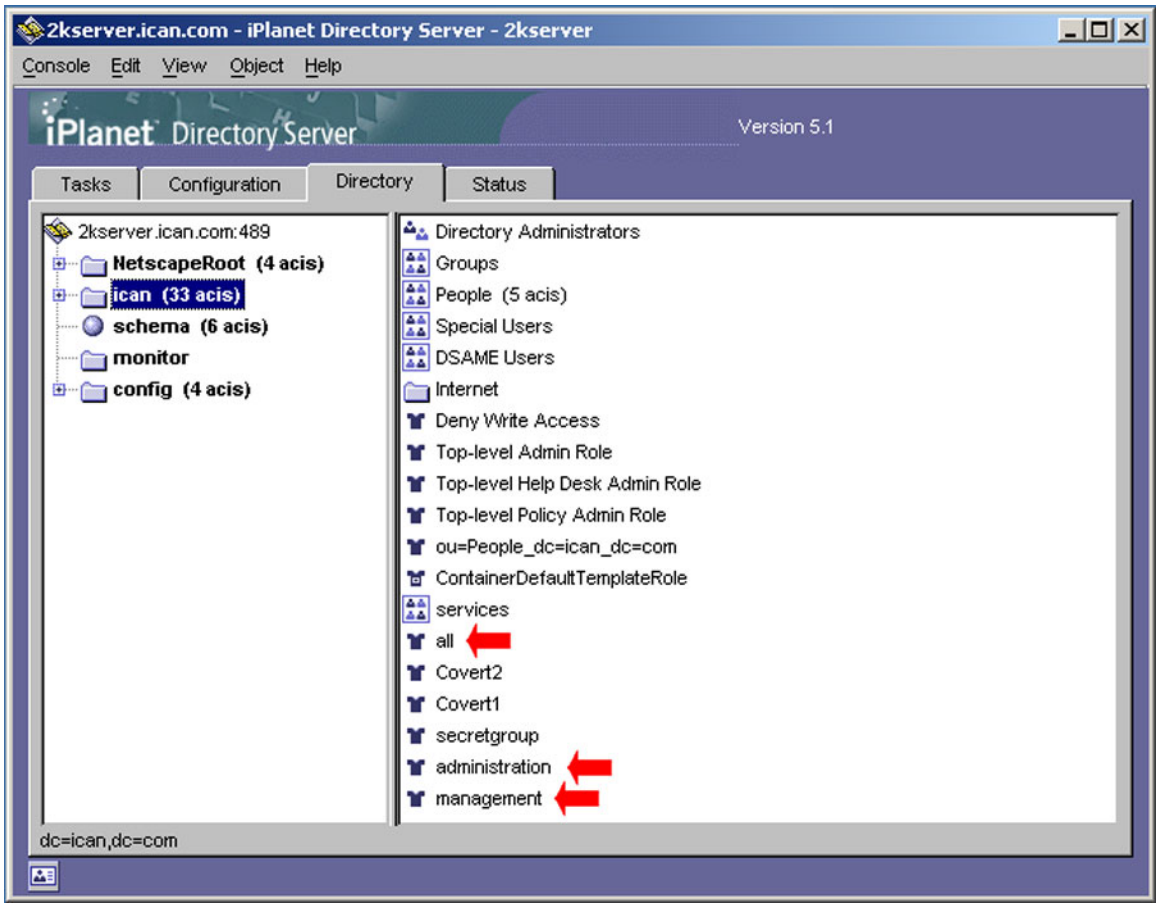
Figure 155 Sun ONE - Create New Role



Note: *"ican.com" is a fictitious URL.*

3 After creating the roles, you will see them listed as shown in Figure 156.

Figure 156 Sun ONE - Roles



Configuring the ICAN Repository to Use LDAP

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 used by the default configuration are shown in Table 30. For a detailed description of all the possible attributes of this Realm class, please refer to the Tomcat documentation for JNDI Realm.

Table 30 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_host	The LDAP server name; for example, 'localhost'.
	LDAP_port	The port your LDAP server listens for requests; for example, 389.
roleBase		The base entry for the role search. If not specified, the search base is the top-level directory context.
roleName		The attribute in a role entry containing the name of that role.
roleSearch		The LDAP search filter for selecting role entries. It optionally includes pattern replacements "**{0}**" for the distinguished name and/or "**{1}**" for the username of the authenticated user.
userBase		The entry that is the base of the subtree containing users. If not specified, the search base is the top-level context.
userPattern		A pattern for the distinguished name (DN) of the user's directory entry, following the syntax supported by the **java.text.MessageFormat** class with **{0}** marking where the actual username should be inserted.
userRoleName		The name of an attribute in the user's directory entry containing zero or more values for the names of roles assigned to this user. In addition you can use the **roleName** property to specify the name of an attribute to be retrieved from individual role entries found by searching the directory. If **userRoleName** is not specified, then all roles for a user derive from the role search.
userRoleNamePattern		A pattern for the distinguished name (DN) of the role's directory entry, following the syntax supported by the **java.text.MessageFormat** class with **{0}** marking the actual role name. This pattern is used to parse the DN to get the actual role name for authorization purposes in ICAN, where the actual username should be inserted.

Attribute	Parameter	Description/Notes
userSearch		The LDAP search filter to use for selecting the user entry after substituting the username in {0}.

Configuring the ICAN Suite for ADS

To add the Realm element for Microsoft Active Directory Server

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

```
<Realm className="org.apache.catalina.realm.JNDIRealm"
               connectionURL="ldap://localhost:389"
               userBase="CN=Users,dc=ican,dc=com"
               userSearch="(cn={0})"
               roleBase="ou=ICANRoles,dc=ican,dc=com"
               roleName="cn"
               roleSearch="(member={0})"
        />
```

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

Configuring the ICAN Suite for SunONE

To add the Realm element for the Sun ONE Directory Server

1. Open *ICAN-root*/repository/server/conf/server.xml.
2. Add the following code snippet inside the **ENGINE** tag:

```
< Realm className="org.apache.catalina.realm.JNDIRealm"
               connectionURL="ldap://localhost:489"
               userBase="CN=People,dc=ican,dc=com"
               userPattern="uid={0},ou=People,dc=ican,dc=com"
               userSearch="(uid={0})"
               userRoleName="nsroledn"
               userRoleNamePattern="cn={0},dc=ican,dc=com"
         />
```

3. Save your changes and close the file.
4. Start your Sun ONE Directory Server.
5. Shut down and restart your ICAN repository server.

10.2.2 Environment User Management

This section describes the management of users who would access the applications deployed in an enterprise, using the ICAN suite. Unlike the Configuration User Management, changes are not dynamic and the Logical Host must be rebooted to reflect the changes.

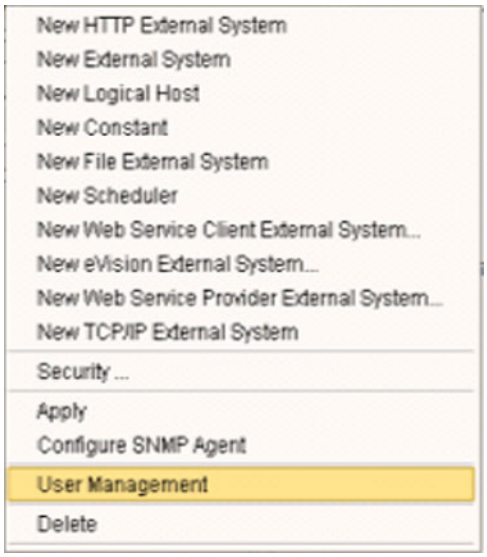
Creating and Configuring Users

When you create an Environment, it comes with one default user: *Administrator*. If you specify any user other than the Administrator in any of your application settings (such as in the Connectivity Map links), you need to create that user in that Environment by right-clicking on the Environment and selecting the **User Management** option.

To Open the User Management Dialog Box

1 In the Environment Explorer, right-click on an **Environment** to display its context menu (see Figure 157).

Figure 157 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 **Configuration User Management** on page 163. Follow the same procedure described in that section.

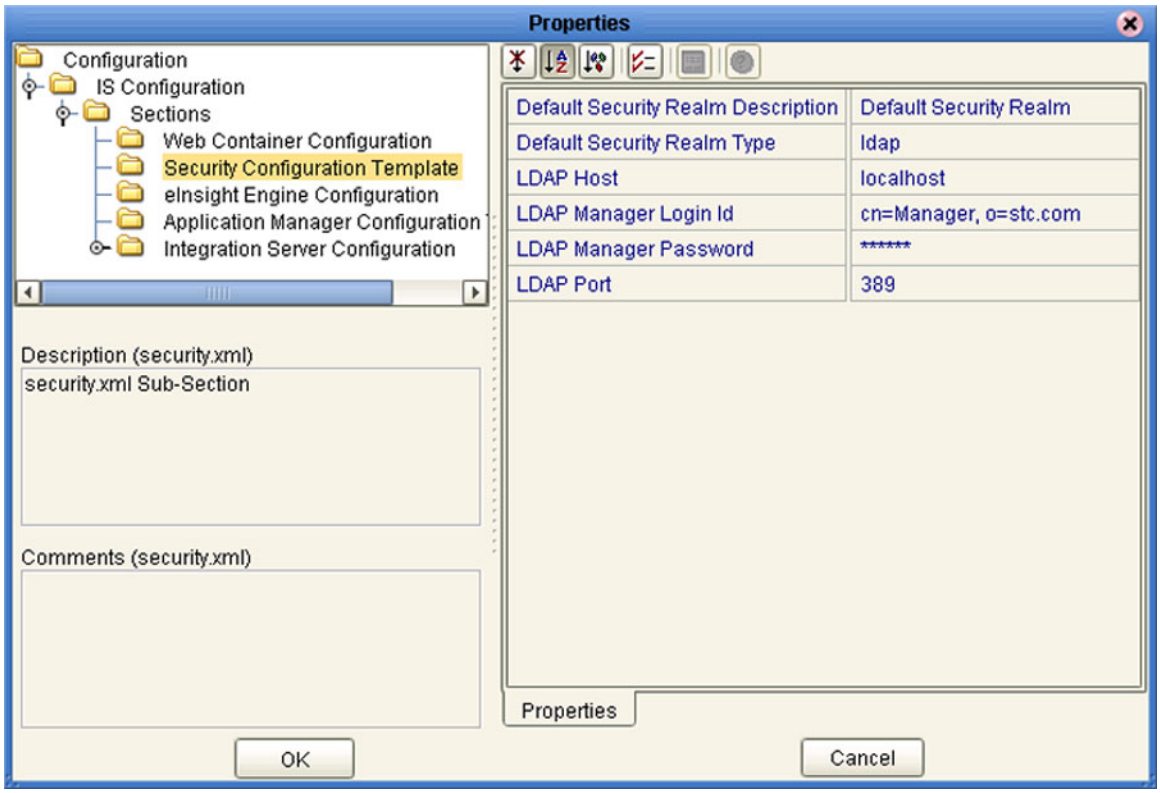
3 Reboot the Logical Host so that the run-time components can pick up the changes.

Configuring for LDAP Servers in Environment 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 158), where:

- **LDAP Host** is the computer 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 158 Security Configuration Template Properties Dialog Box



10.3 ACL Management

When you create any object such as Project, Connectivity Map, or Environment in the Enterprise Designer and store it in the repository, by default no Access Control List (ACL) is set on these objects. Therefore, no permission checks are triggered on these objects when actions are performed involving these objects such as activation or bootstrap. Every repository user will have access to every one of these objects if no ACLs have been specified.

ACL Properties allow you to assign **Read** and/or **Write** access to registered users for a selected object within a Repository. If an object does not have an ACL, all users are authorized to access that object. If an ACL is created for an object, an Administrator is added automatically with **Read** and **Write** permissions.

The actions on a node in Enterprise Designer are enabled or disabled based on the ACL of the repository object associated with the node.

- A user without the **Read** or **Write** permissions will not be able to expand a node to see the children; all the actions on that node will also be disabled.
- A user having only the **Read** permission can expand the node to see the child nodes. The enabling or disabling of the actions on that node will vary, however, based on the type of action. This is based on the ACL of the Repository Object and the Version Control status.

  This logic for this is dependent upon the type of action and the module to which it belongs. For example, the *Delete* action on the Project Elements is disabled if the user does not have the **Write** permission on *both* the Project Element and the parent Project.
- If the user has both the **Read** and **Write** permissions, or if the object does not have any ACL, all the actions on that node are enabled for that user.

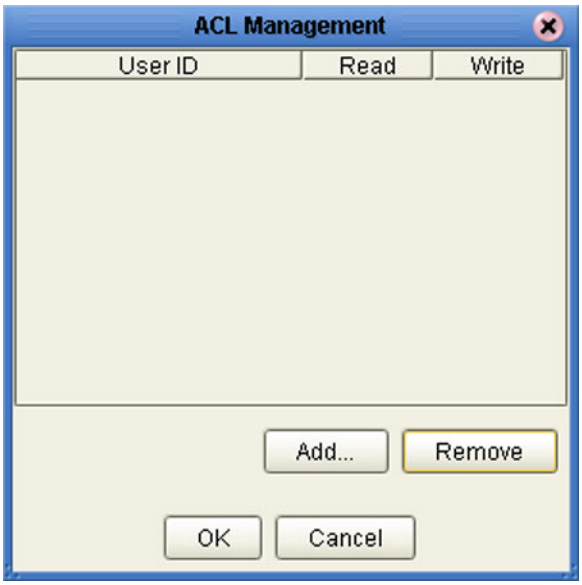
To add and assign access rights to a user

1 From the Project Explorer, select an object icon.

2 Right-click to display the object's context menu.

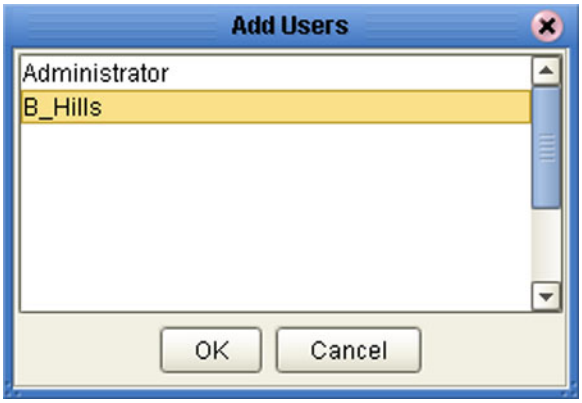
3 Select **ACL Management** to display the dialog box shown in Figure 159.

Figure 159 ACL Management Dialog Box (1)



4 Click **Add** to display the dialog box shown in Figure 160.

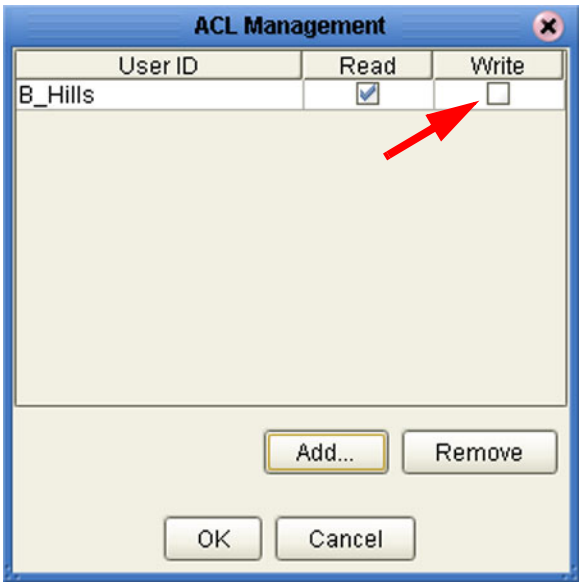
Figure 160 ACL Add Users Dialog Box



5 Select the *existing Repository user* to whom you want to grant access to the object.

6 Click **OK** to add the user to the ACL Management list. The user is automatically assigned **Read** access to the object.

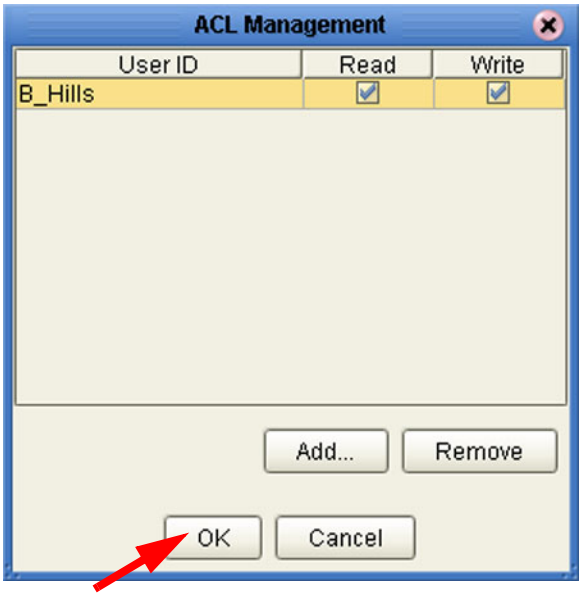
Figure 161 ACL Management Dialog Box (2)



7 Select the **Write** check box for the user if you want them to be able to edit the Project (see Figure 162). This check box can be cleared later if you need to remove **Write** access for this user.

Note: *The Administrator's permissions are preset and cannot be modified.*

Figure 162 ACL Management Dialog Box (3)



8 Click **OK** to save your changes.

Since all ACL users have **Read** access, you cannot clear the **Read** check box; you simply select the user and click **Remove** to delete the user from the ACL for the object.

10.4 ICAN Component Security

10.4.1 Message Server Security

Message server security can be enabled and disabled by modifying the configuration properties for the message server.

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 163).

Figure 163 JMS IQ Manager Properties

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

Note: *See the eGate Integrator JMS Reference Guide for more information on this subject.*

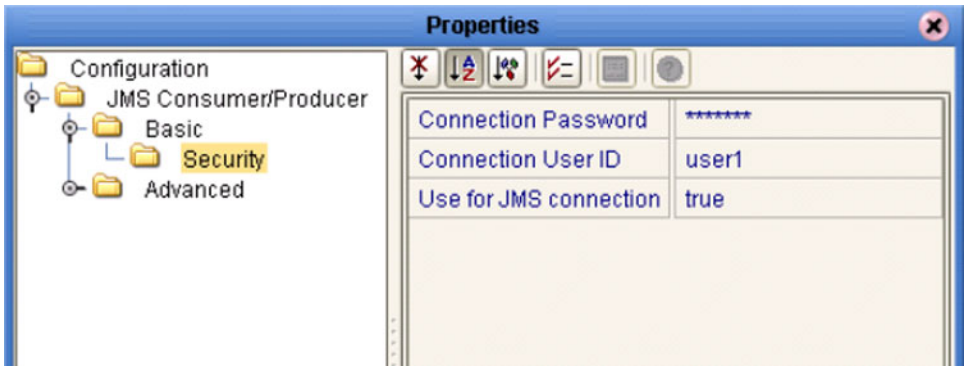
10.4.2 JMS Client Security

Security settings for the generated J2EE applications can be specified by modifying the configuration properties for the JMS Client connections in the Connectivity Map. JMS Client security properties must be specified if security is enabled for the message server.

To enable JMS Client security

To enable authentication of Collaborations in JMS Client connections, set the User ID and Password properties as demonstrated in Figure 164.

Figure 164 JMS Client Security Properties



Note: *See the eGate Integrator JMS Reference Guide for more information on this subject.*

10.5 Using SSL/HTTPS in ICAN

10.5.1 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.

10.5.2 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-root\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.

Important: *You should still have the option to use HTTP to connect to the Enterprise Designer—system administrators should **not** block the HTTP port.*

Chapter 11

Logging

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

11.1 Overview

While a deployment profile is active and running, eGate Integrator automatically generates log messages for the run-time components (Logical Host, Integration Servers, and Message Servers). Enterprise Designer, Enterprise Monitor, and the Repository also have log files, as do several other eGate components. The logging level determines what type of information is recorded, as the logs only contain messages of the specified severity level and higher.

The log message files that are generated for the various eGate components, and their locations, are identified in **Basic Log Files and Locations** on page 202 and **Run-Time Log Files and Locations** on page 206. The corresponding log configuration files, and their configurable properties, are also described.

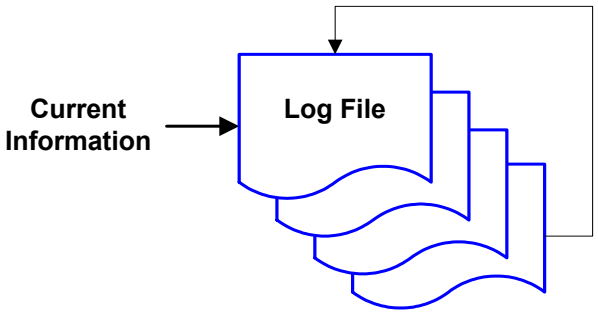
Logs can be viewed and their levels set using the Enterprise Manager, as described in **Viewing Logs** on page 36 and **Setting Log Levels** on page 38.

11.2 eGate Logs

11.2.1 Log File System

While a deployment profile is active and running, eGate Integrator automatically generates log messages for the various runtime components. Other eGate components, such as the Repository, maintain log files whenever they are being used. The log files constitute a recirculating stack (see Figure 165). 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 165 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.

Run-time 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 want to preserve log files (for example, on a weekly basis), you can copy the log files to a backup storage location.

11.2.2 Log File Management

Log files can be managed from the Enterprise Manager, as described in:

- **Viewing Logs** on page 36
- **Setting Log Levels** on page 38

Additional management can be performed from the command line. Locations of all log files, their corresponding configuration files, and configurable properties, are listed in:

- **Basic Log Files and Locations** on page 202
- **Run-Time Log Files and Locations** on page 206

11.2.3 Logging Model

The ICAN logging system is based on the open-source *log4j* API, which is fast, reliable, and extensible, but also relatively simple to use. Log4j has three main components: *Loggers*, *Appenders* and *Layouts*. These three types of components work together to enable the logging of messages according to message type and level, and to allow control (at run time) of how these messages are formatted and where they are reported.

Loggers

The *Logger* is the core component of the logging process, and is responsible for handling the majority of log operations. There are five basic logging levels defined in the log4j API, as shown in the following table.

Table 31 Logging Levels

Event Severity ↑

Level	Designates
FATAL	Very severe error events that will presumably lead eGate to abort.
ERROR	Error events that might still allow eGate to continue running.
WARN	Potentially harmful situations.
INFO	Informational messages that highlight the progress of eGate at a coarse-grained level.
DEBUG	Informational events that are most useful for debug eGate at a fine-grained level. This is the default setting.

↓ Events Logged

A logger will only output messages having a severity level that is higher than or equal to the set level. If the level of a logger is not set, it will default to the DEBUG level, which results in the maximum amount of logging. This situation should be avoided during routine operation because of the negative impact on performance and increased file storage requirements.

Appenders

Appenders are responsible for controlling the output of log operations. Loggers are configured by specifying their Appender properties, as listed in the configuration properties tables. The recirculating stack behavior of the log file system is controlled by the log4j.RollingFileAppender method.

Layouts

Layouts are responsible for formatting the output of the loggers, as displayed in the Monitor (see **Viewing Logs** on page 36).

11.3 Basic Log Files and Locations

11.3.1 Repository

- Master Repository log:

  ICAN-root/repository/logs/repository.log

- Master Repository log configuration file:

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

Table 32 Properties for the Master Repository Log

Property	Default Values
log4j.appender.RepositoryAppender	org.apache.log4j.RollingFileAppender
log4j.appender.RepositoryAppender.File	*ICAN-root*/repository/logs/repository.log
log4j.appender.RepositoryAppender.MaxFileSize	1000KB
log4j.appender.RepositoryAppender.MaxBackupIndex	10
log4j.appender.RepositoryAppender.layout	org.apache.log4j.PatternLayout
log4j.appender.RepositoryAppender.layout.Conversion Pattern	%d{ddMM HH:mm:ss} %5p [%t] - %m%n
log4j.logger.com.stc.repository	DEBUG, RepositoryAppender

- UNIX Repository log:

  ICAN-root/repository/server/logs/catalina.out

- Repository installation log:

  ICAN-root/repository/logs/install.log

- Administration servlet log:

  ICAN-root/repository/server/logs/*hostname*_admin_log.*date*.txt

- Default repository and manifest servlet log:

  ICAN-root/repository/server/logs/*hostname*_log.*date*.txt

- Examples servlet log:

  ICAN-root/repository/server/logs/*hostname*_examples_log.*date*.txt

- Connection log:

  ICAN-root/repository/logs/connection.log

- FTP log:

  ICAN-root/repository/logs/repoftp.log

- UDDI Repository log:

  ICAN-root/repository/logs/stcuddi.log

- UDDI Repository log configuration file:

  ICAN-root/repository/server/webapps/stcuddi/conf/log4j.properties

Table 33 Properties for the UDDI Repository Log

Property	Default Values
log4j.appender.juddilog	org.apache.log4j.RollingFileAppender
log4j.appender.juddilog.File	*ICAN-root*/repository/logs/stcuddi.log
log4j.appender.juddilog.MaxFileSize	10MB
log4j.appender.juddilog.MaxBackupIndex	3
log4j.appender.juddilog.layout	org.apache.log4j.TTCCLayout
log4j.appender.juddilog.layout.ContextPrinting	true
log4j.appender.juddilog.layout.DateFormat	ISO8601
log4j.rootLogger	DEBUG, juddilog

- Deployment Application log:

  ICAN-root/repository/lh-deployment-servlet/deployment-servlet.log

Note: *The Repository Deployment Application log is related to all deployment actions spawned by invoking either the **Apply** menu option from Enterprise Designer or invoking the bootstrap script. If any errors occur during these invocations, this log will contain the root cause of the problem—if the problem originated from the deployment application residing on the repository server.*

11.3.2 ESR Installer Logs

- Log file:

  ICAN-root/esrs.log

- Log configuration file:

  ICAN-root/ESRs/log4j.properties

Table 34 Properties for the ESR Installer Log

Property	Default Values
log4j.appender.juddilog	org.apache.log4j.RollingFileAppender
log4j.appender.juddilog.File	*ICAN-root*/esrs.log
log4j.appender.juddilog.MaxFileSize	10MB
log4j.appender.juddilog.MaxBackupIndex	3
log4j.appender.juddilog.layout	org.apache.log4j.PatternLayout
log4j.appender.juddilog.layout.ConversionPattern	%d{ISO8601} %-5p [%c] %m%n
log4j.rootLogger	DEBUG,File,Console

11.3.3 Enterprise Designer

- Master log file:

  ICAN-root/edesigner/usrdir/system/ide.log

- Master log configuration file:

  ICAN-root/edesigner/bin/log4j.properties

Table 35 Properties for the Enterprise Designer Master Log

Property	Default Values
log4j.appender.R	org.apache.log4j.RollingFileAppender
log4j.appender.R.File	*ICAN-root*/usrdir/system/ide.log
log4j.appender.R.MaxFileSize	1000KB
log4j.appender.R.MaxBackupIndex	100
log4j.appender.R.layout	org.apache.log4j.PatternLayout
log4j.appender.R.layout.ConversionPattern	ICAN5.[%d{DATE}] %p (%F:%L) - %m%n
log4j.rootLogger	DEBUG, R, stdout

11.3.4 Enterprise Monitor

- Enterprise monitor log:

  ICAN-root/monitor/logs/monitor.log

- Log configuration file:

  ICAN-root/monitor/config/log4j.properties

Table 36 Properties for Enterprise Monitor Log

Property	Default Values
log4j.appender.R	org.apache.log4j.RollingFileAppender
log4j.appender.R.File	*ICAN-root*/monitor/logs/monitor.log
log4j.appender.R.MaxFileSize	1000KB
log4j.appender.R.MaxBackupIndex	100
log4j.appender.R.layout	org.apache.log4j.PatternLayout
log4j.appender.R.layout.ConversionPattern	%d %5p [%t] %C (%F:%L) - %m%n
log4j.rootLogger	INFO, R, stdout

11.4 Run-Time Log Files and Locations

Run-time log files, and the directory structure in which they reside, are generated when you bootstrap the Logical Host following deployment.

11.4.1 Logical Host

- Master log file:

  ICAN-root/logicalhost/logs/stc_lh.log

- Master log configuration file:

  ICAN-root/logicalhost/logconfigs/LH/log4j.properties

Table 37 Properties for the Logical Host Log

Property	Default Values
log4j.appender.FILE	org.apache.log4j.RollingFileAppender
log4j.appender.FILE.File	*ICAN-root*/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.*

- Monitor Interface log file

  ICAN-root/logicalhost/logs/stc_ms_stcsysjms.log

11.4.2 Integration Servers

- Log file:

  ICAN-root/logicalhost/logs/stc_is_*int-server-name*.log

- Log configuration file:

  ICAN-root/logicalhost/logconfigs/*int-server-name*/log4j.properties

Table 38 Properties for the Integration Server Logs

Property	Default Values
log4j.appender.FILE	org.apache.log4j.RollingFileAppender
log4j.appender.FILE.File	*ICAN-root*/logicalhost/logs/ stc_is_*int-server-name*.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

11.4.3 Message Servers

- Log file:

  ICAN-root/logicalhost/logs/stc_ms_*msg-server-name*.log

- Log configuration file:

  ICAN-root/logicalhost/stcms/*msg-server-name*/
  stcms.default.properties

Table 39 Properties for Integration Server Logs

Property	Default Values
log4j.appender.FILE	org.apache.log4j.RollingFileAppender
log4j.appender.FILE.File	*ICAN-root*/logicalhost/logs\ stc_ms_*msg-server-name*.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

Glossary

BI
Business integration (also Business Intelligence).

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 211**). 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.

CRM
Customer Relations Management

Data Cleansing
Data must be cleansed of errors in structure and content before it is useful in data warehousing and integration; this means transforming data for accurate and effective use in a database or data management system by cleansing "dirty" or redundant data.

Data Dictionary
Defines the organization of a database and lists all files in the database, the number of records in each file, and the names and types of each field. The data dictionary is often hidden from end users. Although the dictionary doesn't contain actual data, it does contain essential information for managing the database.

Data Integrity
Refers to the accuracy and validity of data. Data integrity can be compromised in many ways, including human error through data entry, or through faulty logic in

programming. Computer viruses, software bugs and many other factors can also compromise data integrity.

Data Mapping
In relational databases (RDBMSs) data mapping is the relationship and data flow between source and target objects. Mapping involves structuring the relationship between source and target objects.

Data Mart
A smaller, focused, database designed to help managers make business decisions. (A data warehouse is a larger, enterprise, database(s).)

Data Mining
Used to synthesize or isolate unique data patterns to predict future behaviors or to filter data to select patterns that help discover previously unknown relationships among data. Commonly used by marketers who acquire and distill consumer information.

Data Transformation
Data transformation is necessary after extracting data from legacy data formats, or any format that requires cleansing. Data is transformed for efficient use for Business-to-Business Enterprise Data Integration.

Data Warehouse
A copy or view of enterprise transaction data (sometimes non-transaction data) that is used for reporting. The data is often summarized and always structured for queries and analysis.

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.

Dimension Table
Dimension tables describe the business entities of an enterprise; also called lookup or reference tables.

Dirty Data
Dirty data contains, but is not limited to, incorrect data including spelling errors, punctuation errors, incorrect data referencing, incomplete, inconsistent, outdated, and redundant data.

Drill Down
To move from summary to more detailed data by “drilling down” to get it. In database terminology this might mean starting with a general category and drilling down to a specific field in a record.

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.

EPR
Enterprise Resource Management

ETL
Extract, Transform, Load. Extract is the process of reading data from a source database and extracting the desired subset of data. Transform is the process of converting the extracted data from its previous form into the desired form. Load is the process of writing the data into a larger database.

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

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

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

Extraction
Data are extracted from a source using software tools. This first step in ETL initially "gets" the data.

Fact Table
A fact table typically contains two types of columns: those containing facts and those that contain foreign keys to dimension tables. Fact tables contain detail facts and/or summary facts.

ICAN Suite
The SeeBeyond Integrated Composite Application Network Suite.

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

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

Join
Matches records, which are joined by a common field, in two tables in a relational database. Often part of a Select query.

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.

Metadata
"Data about data." Metadata describes "how," "when," and "who" about structure and format, of a particular set of data. ETL tools are used to generate and maintain a central metadata repository.

Non-normalized Data
Non-normalized data cannot be cross-referenced accurately, if at all, and causes manageability issues. Non-normalized data may be converted to normalized data.

Normalized Data
Normalization is a common database design process used to remove redundant or incorrect organization and data. The design and normalization of the database will create a maintainable data set that can be cross-referenced.

Normalized data is not only easier to analyze but also easier to expand. Normalization involves removing redundancy and correcting incorrect data structure and organization.

OLAP
Online analytical processing.

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.

Query

A request for information from a database. There are three query methods:

Choose – With this easy-to-use method, the database system presents a list of parameters from which you can choose. This method is not as flexible as other methods.

Query by example (QBE) – With this method, the system lets you specify fields and values to define a query.

Query language – With this method, you have the flexibility and power to make requests for information in the form of a stylized query using a query language. This is the most complex and powerful method.

Queue

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

Raw Data

Data that has not been turned into "information," through processing. Although factual and "real," raw data is unorganized.

Relational Database (RDBMS)

Short for Relational Database Management System, most often referred to as RDBMS. Data is stored in related tables. Relational databases can be viewed in many different ways.

In this system a single database can be spread across several tables. (RDBMS differs from flat-file databases where each database is self-contained as a single file or table.)

Repository

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

Schema Runtime Environment

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

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

Staging Data
Data that is to be processed before entering the warehouse.

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

Table
Refers to data arranged in rows and columns, like a spreadsheet. In relational database management systems, all information is stored in tables.

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

Transformation
Data that are extracted from databases are transformed into a desired form, using various tools that cleanse, merge, purge, aggregate, calculate, audit, remove redundancy, standardize, etc.

XSLT

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

Warehouse
See **"Data Warehouse"**.

e*Gate 4.x Terms in eGate 5.0

Table 40 provides definitions for the terms that are new with eGate release 5.0, as well as equivalent terms from eGate release 4.x.

Table 40 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 40 eGate 5.0 Terms (Continued)

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

Index

A

ACL properties **71–72**, **130**, **132**
Activate button **148**
activating
  Deployment Profile **152**
alerts
  viewing **33**

B

BEA WebLogic **155**
BI **208**
buttons
  Activate **148**
  Deactivate **148**
  Map Variables **148**

C

Collaboration **208**, **214**
  derived **209**
Collaboration definition **208**, **214**
connection **208**, **214**
Connectivity Map **208**, **214**
  Editor **73**
constants **208**
  Environmental **133**
Control Broker **214**
conventions, writing in document **17**
CRM **208**
customizer **80**

D

data cleansing **208**
data dictionary **208**
data integrity **208**
data mapping **209**
data mart **209**
data mining **209**
data transformation **209**
Data Type Definition (DTD) **107**
data warehouse **209**, **213**
Deactivate button **148**
deactivating
  Deployment Profile **152**
deploy **214**
Deployment **214**
Deployment EditorEditors
  Deployment **148**
Deployment Profile **147**, **209**, **214**
  activating **152**
  creating **149**
  deactivating **152**
  map variables **154**
derived Collaboration **209**
dimension table **209**
dirty data **209**
drill down **209**
DTD Wizard **107**

E

e*Way **214**
e*Way Connection **214**
e*Way Connection Configuration **214**
Editor
  Connectivity Map **73**
  OTD **120**
eGate system **210**
enter file name to export **62**
Enterprise Designer **214**
  enterprise explorer **48**
  menu bar **44**
  starting **43**
Enterprise Explorer
  Environments **49**, **126**
  Projects **48**, **68**
Enterprise Manager **214**
  Documentation **30**
  Interface **29**
  starting **28**
Enterprise Monitor **31**, **214**
Environment **23**, **210**, **214**
  constants **133**
Environment Explorer **49**, **126**
EPR **210**
ETD **214**
ETL **210**
Event Type Definition **214**
eWay **210**, **214**
eWay Configuration **214**
external
  application **210**, **214**
  system **210**, **214**
external system **23**
extraction **210**

F

fact table **210**

I

IBM WebSphere **158**
ICAN Suite **210**
impact analyzer
  overview **55**
indoubt transaction **39**
Indoubt transaction (XA) **39**
Integration Server **210**, **214**
Interfaces
  Enterprise Manager **29**

J

JMS
  connection **214**
  e*Way Connection **214**
  IQ Manager **214**
  queue **214**
  topic **215**
JMS IQ Manager **210**
join **210**

L

LDAP
  Configuration User Management **178**
  Environment User Management **189**
  Overview **176**
link **211**, **214**
linked message destination **211**, **214**
Logical Host **23**, **211**, **214**
  starting **144**
  Starting as a Windows Service **141**
  Starting Manually on Windows **145**
  Starting on Linux **145**
  Starting on UNIX **145**
  Startup Configuration File **138**
  stopping **146**
logs
  setting levels **38**
  viewing **36**

M

Management Agent **211**
map variables **154**
Map Variables button **148**
menu bar **44**
message
  destination **211**, **214**
  server **214**
metadata **211**
Monitor
  Enterprise **31**

N

network view **214**
non-normalized data **211**
normalized data **211**

O

Object Type Definition **211**, **214**
  wizard **108**, **113**, **117**
Object Type Definition (OTD) **104**
OLAP **211**
Open File command **121**
OTD **211**, **214**
  Editor **120**
  tester **123**
OTD Editor commands
  Open File **121**
  Refresh OTD **121**
  Run Tester **121**
  Save as New Name **121**
  Save File **121**
  Sort by Name **121**
  Tester **121**
  Toggle Reference Tab Panel **121**
OTD Wizard
  DTD **107**
  WSDL **111**
  XSD **116**

P

Participating Host **214**
Process Manager **214**
Profile, Deployment **147**
Project **212**, **214**
Project Explorer **48**, **68**

Q

query **212**
queue **212**, **214**

R

raw data **212**
rdbm **212**

Refresh OTD command **121**
Registry **214**
relational database **212**
Repository **23**, **212**, **214**
roles **163**
Run Tester command **121**

S

Save as New Name command **121**
Save File command **121**
Scheduler **77**
Schema **214**
Schema Runtime Environment **212**
security **137**
Security Server **213**
select file to import **64**
setting log levels **38**
Sort by Name command **121**
SRE **212**
staging data **213**
starting
  Enterprise Designer **43**
  Enterprise Manager **28**
subproject **213–214**
supporting documents **18**

T

table **213**
tester
  OTD **123**
Tester command **121**
Toggle Reference Tab Panel command **121**
topic **213–215**
transformation **213**

V

variables
  mapping **154**
version control **57**
viewing
  alerts **33**
  logs **36**

W

warehouse **213**
WebLogic **155**
WebSphere **158**
Wizard
  OTD (DTD) **107**
  OTD (WSDL) **111**
  OTD (XSD) **116**
WSDL Wizard **111**

X

XA transaction
  Indoubt **39**
XA transactions **39**
XSD Wizard **116**
XSLT **213**, **215**