

***SeeBeyond ICAN Suite***

# eGate Integrator for eInsight Enterprise Service Bus User's Guide

*Release 5.0.5*



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

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

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 This work is protected as an unpublished work under the copyright laws.

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

Version 20041119144744.

# Contents

<a href="#">List of Figures</a>	<a href="#">9</a>
---------------------------------	-------------------

<a href="#">List of Tables</a>	<a href="#">14</a>
--------------------------------	--------------------

## Chapter 1

<a href="#">Introduction</a>	<a href="#">16</a>
<a href="#">About eGate Integrator</a>	<a href="#">16</a>
<a href="#">What's New in This Release</a>	<a href="#">17</a>
<a href="#">LDAP Support</a>	<a href="#">17</a>
<a href="#">Enterprise Designer</a>	<a href="#">17</a>
<a href="#">Basic User Interface</a>	<a href="#">17</a>
<a href="#">Project Deployment and Activation</a>	<a href="#">17</a>
<a href="#">Version Control</a>	<a href="#">18</a>
<a href="#">Web Services</a>	<a href="#">18</a>
<a href="#">About This User's Guide</a>	<a href="#">19</a>
<a href="#">What's in This User's Guide</a>	<a href="#">19</a>
<a href="#">Scope</a>	<a href="#">19</a>
<a href="#">Intended Audience</a>	<a href="#">20</a>
<a href="#">Document Conventions</a>	<a href="#">20</a>
<a href="#">Screenshots</a>	<a href="#">20</a>
<a href="#">Related Documents</a>	<a href="#">21</a>
<a href="#">The SeeBeyond Web Site</a>	<a href="#">21</a>
<a href="#">SeeBeyond Documentation Feedback</a>	<a href="#">21</a>

## Chapter 2

<a href="#">System Overview</a>	<a href="#">22</a>
<a href="#">Introduction</a>	<a href="#">22</a>
<a href="#">Integration Model</a>	<a href="#">23</a>
<a href="#">System Architecture</a>	<a href="#">25</a>
<a href="#">Repository</a>	<a href="#">26</a>
<a href="#">Run-Time Environments</a>	<a href="#">26</a>
<a href="#">User Interfaces</a>	<a href="#">27</a>

Enterprise Manager	27
Enterprise Designer	28

## Chapter 3

<b>Enterprise Manager</b>	<b>30</b>
<b>Overview</b>	<b>30</b>
Installing and Updating eInsight Enterprise Service Bus	30
Monitoring and Managing eInsight Enterprise Service Bus	30
<b>Starting Enterprise Manager</b>	<b>31</b>
<b>The Enterprise Manager Interface</b>	<b>32</b>
Home	33
Documentation	34
<b>The ICAN Monitor</b>	<b>35</b>

## Chapter 4

<b>Enterprise Designer</b>	<b>36</b>
<b>Overview</b>	<b>36</b>
User Interface	36
Editors	37
Analysis and Archiving Tools	37
<b>Using a Proxy Server</b>	<b>38</b>
<b>Starting Enterprise Designer</b>	<b>40</b>
<b>Interface Features</b>	<b>42</b>
<b>Menus</b>	<b>42</b>
File Menu	42
Tools Menu	42
View Menu	42
Window Menu	43
Help Menu	43
Options Setup	44
<b>Toolbar</b>	<b>45</b>
<b>Browser Buttons</b>	<b>46</b>
<b>Enterprise Explorer</b>	<b>47</b>
Project Explorer	47
Environment Explorer	48
<b>Enterprise Designer Editors</b>	<b>49</b>
Connectivity Map Editor	49
OTD Editor	50
Environment Editor	51
Deployment Editor	52
<b>Additional Tools and Features</b>	<b>53</b>
Project/Environment Import	53

Importing a Project Using Enterprise Designer	53
Importing a Project Using the Command Line	56
<b>Project/Environment Export</b>	<b>57</b>
Exporting a Project Using Enterprise Designer	57
Exporting a Project Using the Command Line	61
<b>Impact Analyzer</b>	<b>62</b>
<b>Version Control</b>	<b>64</b>
Viewing a Component’s Version History	65
Checking a Component In	66
Checking a Component In Without Revisions	67
Checking a Component Out	68
Retrieving a Component to Your Workspace	69
Checking In a Previous Version as the Latest Version	71
Command-line Utilities	72

## Chapter 5

<b>ICAN Projects</b>	<b>73</b>
<b>Overview</b>	<b>73</b>
Project Components	73
<b>Project Explorer</b>	<b>74</b>
Project Explorer Icons	75
Context Menus	76
Repository Menu	76
Project Menu	77
Connectivity Map Menu	79
Deployment Profile Menu	80
File Menu	81
Object Type Definition Menu	82
<b>Creating a Project</b>	<b>84</b>
<b>Using the Connectivity Map Editor</b>	<b>86</b>
<b>External Applications</b>	<b>88</b>
Schedulers	89
Time Zone	89
Schedule Type	89
Schedule Value	90
<b>Component Connections</b>	<b>91</b>
Configuring a Connection	92
<b>Defining Constants and Variables</b>	<b>94</b>
Project Constants	94
Project Variables	95
Variables & Constants Object Group	96

---

**Chapter 6**

<b>Object Type Definitions</b>	<b>97</b>
<b>Overview</b>	<b>97</b>
<b>Using the OTD Wizards</b>	<b>98</b>
<b>OTD Editor</b>	<b>100</b>
OTD Tester	102
Using the OTD Tester	104
<b>Creating Externally-Defined OTDs</b>	<b>107</b>
Using the DTD Wizard	107
Using the WSDL Wizard	112
WSDL OTD Structure	116
WSDL Operation Elements	116
SOAP Header Elements	117
Using the XSD Wizard	118

---

**Chapter 7**

<b>Environments</b>	<b>123</b>
<b>Overview</b>	<b>123</b>
<b>Environment Explorer</b>	<b>124</b>
Environment Explorer Icons	125
Context Menus	126
Repository Menu	126
Environment Menu	127
Logical Host Menu	128
Integration Server	130
SeeBeyond JMS IQ Manager	131
<b>Environment Editor</b>	<b>132</b>
Defining Environmental Constants	132
<b>Logical Hosts</b>	<b>134</b>
Overview	134
Bootstrapping	135
Configuring a Logical Host	136
Configuring the Base Port Number	137
<b>Integration Servers</b>	<b>139</b>
Configuring an Integration Server	139
Web Container Configuration	141
Web Server Configurations	141
Default Web Server	142
Performance Monitoring (Profiling) Configuration	144
Security Realm Configuration	145
Security Realm Configuration: Sun Java System	146
Security Realm Configuration: Active Directory Service	148
Security Realm Configuration: OpenLDAP Directory Server	150
eInsight Engine Configuration	151

Application Manager Configuration	152
Integration Server Configuration	153
JDBC DataSource Connection Pools	155
Oracle JDBC Connection Pool	155
Deploying User-Defined Stateless Session Beans	157
<b>Message Servers</b>	<b>160</b>
SeeBeyond JMS IQ Manager Configuration	160
General Configuration	160
Performance	162
Messaging Behavior	164
Sun Java System	165
Active Directory Service	167
Diagnostics	169
Stable Storage	171
JNDI	172
OpenLDAP Directory Server	173

---

## Chapter 8

<b>Project Deployment</b>	<b>174</b>
Deployment Profiles	174
Deployment Editor	175
Creating a Deployment Profile	176
Automapping	179
Mapping Variables	180
Activating and Deactivating Deployment Profiles	182
Using Enterprise Designer	182
Using a Command-line Script	184
Deploying Projects to Third-Party Servers	185
BEA WebLogic	185
IBM WebSphere	188

---

## Chapter 9

<b>Web Services</b>	<b>191</b>
Overview	191
SeeBeyond Web Services	192
UDDI Registry	193
Using UDDI Browsers	195
Web Services Application	196
Web Services External System	197
Load Balancing	199
Configuring the Apache Server	199

## Contents

Creating the Mapping File	200
Adding the Rewrite Rule	200
Debugging	201
<b>Building a Web Service</b>	<b>202</b>
Procedure	202
Web Server Example	203
Object Type Definition	203
eInsight Business Process	204
Project Mapping	205
Project Deployment	206
<b>Building a Web Services Client</b>	<b>207</b>
Procedure	207
Web Client Example	208
Object Type Definition	208
eInsight Business Process	210
Project Mapping	211
Project Deployment	212
Project Operation	213
<b>Setting Up Secure Web Services</b>	<b>214</b>
SSL Overview	214
Certificates and Keys	214
Keytool Utility	215
Web Services Setup	216
Web Service Setup	216
Web Services Client Setup	217
Web Services Example Project	218
Web Service Example	218
Web Services Client Example	221
Project Operation	223
Creating Keystores and Self-Signed Certificates	224

---

## Chapter 10

<b>Troubleshooting</b>	<b>225</b>
Error Messages	225
OutOfMemory	225
<b>Glossary</b>	<b>226</b>
<b>Index</b>	<b>232</b>



# List of Figures

Figure 1	eGate Integrator	23
Figure 2	eGate Integrator Implementation Model	24
Figure 3	Distributed eGate Integrator System	25
Figure 4	SeeBeyond Enterprise Manager	27
Figure 5	Enterprise Designer	28
Figure 6	Connectivity Map Editor	29
Figure 7	Enterprise Manager Login	31
Figure 8	Enterprise Manager GUI	32
Figure 9	ICAN Monitor Launch Icon	33
Figure 10	Documentation Page	34
Figure 11	ICAN Monitor Interface - Initial	35
Figure 12	ICAN Monitor Interface - Environment	35
Figure 13	Enterprise Designer User Interface	37
Figure 14	Update Center Wizard	38
Figure 15	Proxy Configuration Dialog Box	39
Figure 16	Login Dialog Box	40
Figure 17	Options Setup - Heap Size Dialog	44
Figure 18	Options Setup - Language Dialog	44
Figure 19	Enterprise Explorer: Project Explorer View	47
Figure 20	Enterprise Explorer: Environment Explorer View	48
Figure 21	Connectivity Map Editor	49
Figure 22	OTD Editor	50
Figure 23	Environment Editor	51
Figure 24	Deployment Editor	52
Figure 25	Import Message Box	53
Figure 26	Import Manager Dialog Box (1)	54
Figure 27	Open File Dialog Box	54
Figure 28	Import Manager Dialog Box	55
Figure 29	Import Status Message Box	55
Figure 30	Export Manager Dialog Box (1a)	57
Figure 31	Export Manager Dialog Box (1b)	58
Figure 32	Export Manager Dialog Box (2)	59

## List of Figures

Figure 33	Save As Dialog Box	59
Figure 34	Enter File Name Dialog Box (2)	60
Figure 35	Export Status Message Box	60
Figure 36	Impact Analyzer Dialog Box	62
Figure 37	<i>Checked In</i> Icon (OTD Example)	64
Figure 38	<i>Checked Out</i> Icon	64
Figure 39	<i>Retrieved</i> Icon	64
Figure 40	Version Control - History Dialog Box	65
Figure 41	Version Control - Check In Dialog Box	66
Figure 42	Version Control - Undo Check Out Dialog Box	67
Figure 43	Version Control - Check Out Dialog Box	68
Figure 44	Version Control - History Dialog Box	69
Figure 45	Access File Dialog Box	70
Figure 46	Confirm Version Replace Dialog Box	70
Figure 47	Make Latest Dialog Box	71
Figure 48	Confirm Latest Version Override Dialog Box	72
Figure 49	Project Explorer	74
Figure 50	Repository Menu	76
Figure 51	Project Menu	77
Figure 52	Connectivity Map Menu	79
Figure 53	Deployment Profile Menu	80
Figure 54	File Menu	81
Figure 55	OTD Menu	82
Figure 56	Repository Context Menu - New Project	84
Figure 57	Project Context Menu - New Project	84
Figure 58	Connectivity Map Window	86
Figure 59	Linking Multiple Message Destinations	87
Figure 60	External Application Drop-Down Menu	88
Figure 61	Scheduler Properties Dialog Box - Time Zone	89
Figure 62	Scheduler Properties Dialog Box - Schedule Type	90
Figure 63	Scheduler Properties Dialog Box - Schedule Value	90
Figure 64	Connection Icons in a Connectivity Map	91
Figure 65	Default Configuration Dialog Box	92
Figure 66	Creating a Project Constant	94
Figure 67	Creating a Project Variable	95
Figure 68	Variables and Constants Object Group	96
Figure 69	OTD Wizard Selection Dialog	98
Figure 70	OTD Editor	100

## List of Figures

Figure 71	OTD Tester	102
Figure 72	Select Data File	104
Figure 73	Object Elements and Values	104
Figure 74	Status Data Display	105
Figure 75	Verbose Data Display	105
Figure 76	OTD Tester Node Table	106
Figure 77	Serialized Data in Output Panel	106
Figure 78	OTD Wizard Selection: DTD Wizard	107
Figure 79	Select DTD File(s) Dialog Box	108
Figure 80	Cannot Create OTD Warning Box	108
Figure 81	Select Document Elements Dialog Box	109
Figure 82	Select OTD Options Dialog Box	110
Figure 83	OTD Wizard Selection: WSDL Wizard	112
Figure 84	WSDL Wizard: Select WSDL Location	113
Figure 85	WSDL Wizard: Select WSDL File	114
Figure 86	WSDL Wizard: Select OTD Options	115
Figure 87	Typical WSDL OTD Structure	116
Figure 88	Output Message with SOAP Header Included	117
Figure 89	OTD Wizard Selection: XSD Wizard	118
Figure 90	XSD Wizard: Select XSD File(s)	119
Figure 91	Cannot Create OTD Warning Box	119
Figure 92	Select Document Elements Dialog Box	120
Figure 93	Select OTD Options Dialog Box	121
Figure 94	Enterprise Explorer: Environment Explorer View	124
Figure 95	Repository Menu	126
Figure 96	Environment Menu	127
Figure 97	Logical Host Menu	128
Figure 98	Logical Host Menu with Third-Party Servers	129
Figure 99	Integration Server Menu	130
Figure 100	JMS IQ Manager Menu	131
Figure 101	Environment Editor	132
Figure 102	Environmental Constants Panel	133
Figure 103	Logical Hosts	134
Figure 104	Startup Sequence	135
Figure 105	Logical Host Configuration Properties	136
Figure 106	Management Agent Configuration Properties	137
Figure 107	Top-level IS Configuration Properties	139
Figure 108	Web Container Configuration Properties	141

Figure 109	Default Web Server Properties	142
Figure 110	Profiling Configuration Properties	144
Figure 111	Security Configuration Properties	145
Figure 112	Sun Java System Configuration Properties	146
Figure 113	Active Directory Server Configuration Properties	148
Figure 114	OpenLDAP Directory Server Configuration Properties	150
Figure 115	eInsight Engine Configuration Properties	151
Figure 116	Application Manager Configuration Properties	152
Figure 117	Integration Server Configuration Properties	153
Figure 118	Oracle JDBC Connection Pool Properties	155
Figure 119	JMS IQ Manager - General Configuration Properties	160
Figure 120	Performance Configuration Properties	162
Figure 121	Messaging Behavior Configuration Properties	164
Figure 122	Sun Java System Configuration Properties	165
Figure 123	Active Directory Service Configuration Properties	167
Figure 124	Diagnostics Configuration Properties	169
Figure 125	Stable Storage Configuration Properties	171
Figure 126	OpenLDAP Directory Server Configuration Properties	173
Figure 127	Deployment Editor Window	175
Figure 128	Web Client Example Project	176
Figure 129	Web Client Example Environment	177
Figure 130	Example Deployment Profile (1)	177
Figure 131	Example Deployment Profile (2)	178
Figure 132	Example Deployment Profile (3)	178
Figure 133	Auto Map Icon	179
Figure 134	Automap Results Dialog Box	179
Figure 135	Automap Options Dialog Box	179
Figure 136	Map Variables Button	180
Figure 137	Deployment Profile Mappings	180
Figure 138	Project Variable Value Entry	181
Figure 139	Activate Dialog Box	182
Figure 140	Success Information Box	182
Figure 141	Logical Host Context Menu - Apply	183
Figure 142	Deactivate Dialog Box	183
Figure 143	WebLogic Deployment (1)	186
Figure 144	WebLogic Deployment (2)	186
Figure 145	WebLogic Deployment Verification	187
Figure 146	WebSphere Deployment (1)	188

## List of Figures

Figure 147	WebSphere Deployment (2)	189
Figure 148	WebSphere Deployment Verification	190
Figure 149	SeeBeyond UDDI Registry	193
Figure 150	Example Web Service WSDL File	193
Figure 151	Microsoft Visual Studio Example	194
Figure 152	Enterprise Designer - Connectivity Map Editor	196
Figure 153	Web Service External System Properties Dialog Box	197
Figure 154	Load Balancing Example	199
Figure 155	Development Flow	202
Figure 156	Wizard Step 3 - Select WSDL File	203
Figure 157	Wizard Step 4 - Options	204
Figure 158	Web Service Business Process	204
Figure 159	Web Service Example Project - Explorer Tree	205
Figure 160	Web Service Example Project - Business Process Mapping	205
Figure 161	Web Service Example Project - Connectivity Map	205
Figure 162	Web Server Example Project - Deployment	206
Figure 163	Development Flow	207
Figure 164	Example Web Service WSDL File	208
Figure 165	SeeBeyond UDDI Browser	208
Figure 166	Select WSDL File	209
Figure 167	Wizard Step 4 - Options	209
Figure 168	Web Client Business Process	210
Figure 169	Web Client Business Process <i>Receive</i> Rule	210
Figure 170	Web Client Business Process <i>Write</i> Rule	210
Figure 171	Web Services Client Example Project - Explorer Tree	211
Figure 172	Web Services Client Example Project - Business Process Mapping	211
Figure 173	Web Services Client Example Project - Connectivity Map	212
Figure 174	Web Services Client Example Project - Deployment	212
Figure 175	Example Secure Web Services Project - Explorer Tree	218
Figure 176	Example Secure Web Services Project - Business Process	219
Figure 177	Example Secure Web Services Project - Connectivity Map	219
Figure 178	Example Secure Web Services Client Project - Explorer Tree	221
Figure 179	Example Secure Web Services Client Project - Business Process	221
Figure 180	Example Secure Web Services Client Project - Business Process Mapping	222
Figure 181	Example Secure Web Services Client Project - Connectivity Map	222

# List of Tables

Table 1	Writing Conventions	20
Table 2	Enterprise Manager - Pages	32
Table 3	Enterprise Manager - Control Tabs	32
Table 4	Document Categories	34
Table 5	ICAN Monitor Interface - Details Tabs	35
Table 6	File Menu Options	42
Table 7	Tools Menu Options	42
Table 8	View Menu Options	42
Table 9	Window Menu Options	43
Table 10	Help Menu Options	43
Table 11	Enterprise Designer Toolbar Icons	45
Table 12	Browser Buttons	46
Table 13	Impact Analyzer Command Buttons	63
Table 14	Project Icons	75
Table 15	Repository Menu Options	76
Table 16	Project Menu Options	77
Table 17	Connectivity Map Menu Options	79
Table 18	Deployment Profile Menu Options	80
Table 19	File Menu Options	81
Table 20	OTD Menu Options	82
Table 21	Connectivity Map Toolbar Icons	87
Table 22	Configuration Dialog Box Toolbar Buttons	93
Table 23	Project Variable Options	94
Table 24	Project Variable Options	95
Table 25	OTD Wizard Navigation Buttons	99
Table 26	OTD Editor Toolbar Icons	101
Table 27	OTD Tester Buttons	103
Table 28	OTD Tester Icons	103
Table 29	DTD OTD Options	110
Table 30	XSD OTD Options	121
Table 31	Environment Icons	125
Table 32	Repository Menu Options	126

Table 33	Environment Menu Options	127
Table 34	Logical Host Menu Options	128
Table 35	Integration Server Menu Options	130
Table 36	Integration Server Menu Options	131
Table 37	Environmental Constants Panel Icons	133
Table 38	Logical Host Configuration Properties List	136
Table 39	Management Agent Configuration Properties List	138
Table 40	Top-level IS Configuration Properties List	140
Table 41	Web Container Configuration Properties List	141
Table 42	Default Web Server Properties List	142
Table 43	Profiling Configuration Properties List	144
Table 44	Security Realm Configuration Properties List	145
Table 45	Sun Java System Configuration Properties List	146
Table 46	Active Directory Server Configuration Default Properties List	148
Table 47	OpenLDAP Directory Server Configuration Default Properties List	150
Table 48	Application Manager Configuration Properties List	152
Table 49	Integration Server Configuration Properties List	153
Table 50	Oracle JDBC Connection Pool Properties List	155
Table 51	JMS IQ Manager - General Configuration Properties List	161
Table 52	Performance Configuration Properties List	162
Table 53	Messaging Behavior Configuration Properties List	164
Table 54	Sun Java System Configuration Default Properties List	165
Table 55	Active Directory Service Configuration Default Properties List	167
Table 56	Diagnostics Configuration Properties List	169
Table 57	Stable Storage Configuration Properties List	171
Table 58	OpenLDAP Directory Server Configuration Default Properties List	173
Table 59	Deployment Toolbar Buttons	175
Table 60	UDDI Registry Information	195
Table 61	Web Services External System Properties	197

# Introduction

This chapter describes the new features in this release of eGate Integrator for eInsight Enterprise Service Bus, the scope and organization of this document, and provides references to additional sources of relevant information.

## What's in This Chapter

- [About eGate Integrator](#) on page 16
- [What's New in This Release](#) on page 17
- [About This User's Guide](#) on page 19
- [What's in This User's Guide](#) on page 19
- [Related Documents](#) on page 21
- [The SeeBeyond Web Site](#) on page 21
- [SeeBeyond Documentation Feedback](#) on page 21

---

## 1.1 About eGate Integrator

SeeBeyond® eGate™ Integrator is an advanced, distributed integration platform that serves as the foundation of the SeeBeyond Integrated Composite Application Network™ (ICAN™) Suite. The user interfaces—Enterprise Designer for Project design and Enterprise Manager for browser-based installation, management, and monitoring—ensure a unified look and feel across all editors in the suite, with a single sign-on process for access to any product.

During Project design, projects are made more maintainable by separating views of logical components from views of the physical environment—the Project Explorer view and the Environment Explorer view.

The run-time environment, which is J2EE-compatible and certified, features high performance and dynamic scalability. The architecture uses EJB (Enterprise Java Beans) with JMS (Java Message Service) and JNDI (Java Naming and Directory Interface). Using JCA (Java Connector Architecture), configurations are deployable to J2EE-compliant implementations, such as WebLogic 8.1 and WebSphere 5.



---

## 1.2 What's New in This Release

For information regarding operating system support, language support, and performance improvements, please refer to the *eInsight Enterprise Service Bus Installation Guide* and *eGate Integrator Release Notes*.

### 1.2.1 LDAP Support

- eInsight Enterprise Service Bus now supports OpenLDAP Directory Server as well as Microsoft's Active Directory and Sun Microsystems's Sun Java System Directory Server. In addition, LDAPS (LDAP over SSL) is supported for encrypted communications between the Repository and all three supported directory servers.
- To support connections to LDAP servers that store information in a hierarchical structure, new sub-tree search properties were added to the LDAP server Properties sections of the Integration Server Properties and JMS IQ Manager Properties dialogs.

### 1.2.2 Enterprise Designer

#### Basic User Interface

- In Project Explorer and applicable editors, the default *SeeBeyond* folder now has a unique icon to distinguish it from all other projects.
- In Environment Explorer, the context menu items under *New...* (such as the list of external systems) are now sorted alphabetically.
- The extended language option now supports additional character sets for Japanese, Korean, and Chinese locales.
- The *Save As* command now allows saving an OTD to a different project.

#### Project Deployment and Activation

- In the Deployment Editor, the new *Automap* feature automatically assigns components to their matching containers when there is a one-to-one correspondence between them. This feature only works with external systems for which it is enabled. To find out if Automap is enabled for a specific component, see the user documentation for that component.
- A warning prompt for deactivation is now issued before deleting a Project that has one or more active deployment profiles. (All active deployment profiles must be deactivated before the project can be deleted.)

## Version Control

- The *Undo Checkout*, *Make Latest*, and *Version History: Retrieve Previous Version* options are now available for all eGate components, except WSDL OTDs.

## Web Services

- Web Services and Web Services Clients now can support the Secure Sockets Layer (SSL). New properties have been added to the following configuration dialogs:
  - ♦ Integration Server Properties/Default Web Server
  - ♦ WebService External System

---

## 1.3 About This User's Guide

### 1.3.1 What's in This User's Guide

- **Chapter 1 "Introduction"** describes the purpose of this User's Guide, including writing conventions and a list of related documents.
- **Chapter 2 "System Overview"** provides an overview of the general structure, architecture, and operation of eInsight Enterprise Service Bus, and its place within the SeeBeyond ICAN Suite.
- **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 "ICAN 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 "Object Type Definitions"** describes how to create Object Type Definitions (OTDs).
- **Chapter 7 "Environments"** explains how to create and populate eInsight Enterprise Service Bus Environments, and how to configure and start Logical Hosts.
- **Chapter 8 "Project Deployment"** explains how to create and activate Deployment Profiles.
- **Chapter 9 "Web Services"** describes how to use eInsight Enterprise Service Bus in concert with other ICAN Suite components to create Web services.
- **Chapter 10 "Troubleshooting"** contains tips for troubleshooting the eGate Integrator part of eInsight Enterprise Service Bus.

In addition, the [Glossary](#) on page 226 lists various terms used in this User's Guide.

### 1.3.2 Scope

This User's Guide provides general information about the features and operation of eInsight Enterprise Service Bus in creating and deploying ICAN Projects. For information on system management, see the *eGate Integrator System Administration Guide*.

**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 ICAN Projects.*

### 1.3.3 Intended Audience

This User's Guide is intended for personnel who are involved in integrating software applications using eInsight Enterprise Service Bus. To a large extent, these are individuals who will be using the Enterprise Designer to build ICAN Projects to accomplish this task. This guide also provides a basic overview of the eGate Integrator part of the product for those attempting to gain a general understanding of eGate.

This guide assumes that the reader is an experienced computer user, familiar with Windows-style GUI operations, and also has an in-depth understanding of the operating system(s) on which eInsight Enterprise Service Bus will be installed.

### 1.3.4 Document Conventions

The following conventions are observed throughout this document.

**Table 1** Writing Conventions

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

### 1.3.5. Screenshots

Depending on what products you have installed, and how they are configured, the screenshots in this document may differ from what you see on your system.

---

## 1.4 Related Documents

The following SeeBeyond documents provide additional information about the eInsight Enterprise Service Bus system as explained in this guide:

- *eGate Integrator JMS Reference Guide*
- *eGate Integrator Release Notes*
- *eGate Integrator System Administration Guide*
- *eGate Integrator Tutorial*
- *eInsight Enterprise Service Bus Installation Guide*
- *eInsight Enterprise Service Bus User's Guide*

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 SeeBeyond documentation is included in the *SeeBeyond ICAN Suite Primer*.

The documentation for the SeeBeyond ICAN Suite is distributed as a collection of online documents, which can be accessed through the Enterprise Manager (see **Documentation** on page 34). These documents are in Adobe Acrobat format, which requires that Acrobat Reader be installed on your computer. Acrobat Reader can be from Adobe Systems as a free download from the following URL:

<http://www.adobe.com>

---

## 1.5 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>

---

## 1.6 SeeBeyond Documentation Feedback

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

[docfeedback@seebeyond.com](mailto:docfeedback@seebeyond.com)

# System Overview

This chapter provides an overview of the conceptual operation and general architecture of the eGate Integrator part of eInsight Enterprise Service Bus.

## What's in This Chapter

- **Introduction** on page 22
- **Integration Model** on page 23
- **System Architecture** on page 25
- **User Interfaces** on page 27

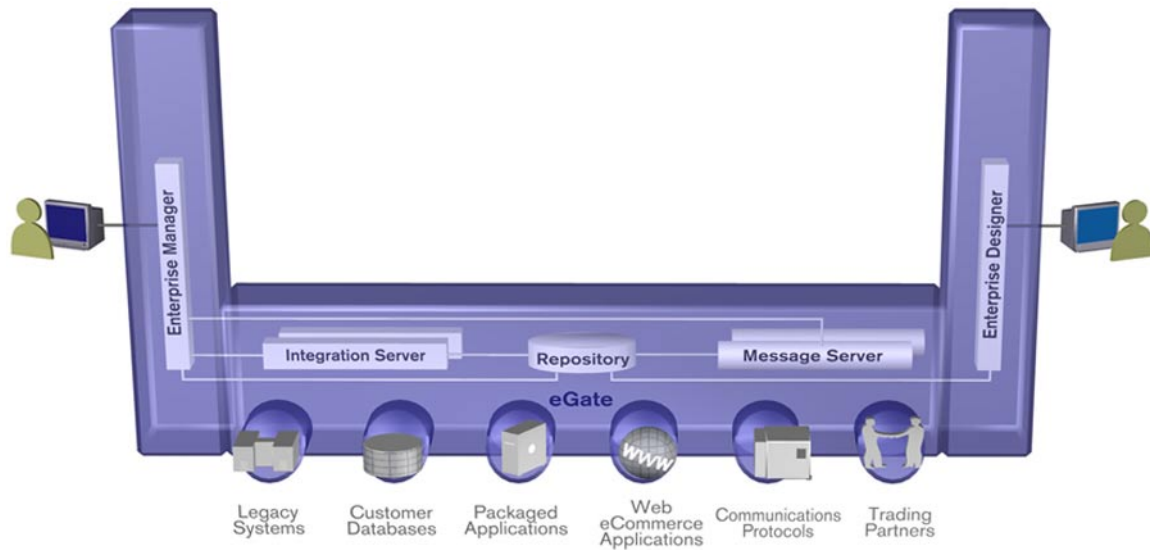
---

## 2.1 Introduction

SeeBeyond® eGate™ Integrator is an advanced, distributed integration platform that serves as the foundation of the SeeBeyond Integrated Composite Application Network™ (ICAN™) Suite. It provides the core integration function, incorporating comprehensive systems connectivity, guaranteed messaging, and robust transformation capabilities. eGate Integrator also provides a unified, single sign-on environment for integration development, deployment, monitoring and management. eGate Integrator supports portability of integrations across common J2EE application servers through a completely open, J2EE-certified and Web Services-based architecture.

As shown in Figure 1, the heart of eGate Integrator is the Repository, which is a comprehensive store of information common to the entire enterprise. An integrated UDDI registry allows publication and discovery of Web services. The run-time environment employs J2EE-compatible integration servers as operational engines and JMS-compatible message servers for the propagation of messages. The flexibility of the eGate Integrator system allows the option of deployment to an ICAN run-time environment or to third-party application servers, across a distributed network of hardware platforms.

**Figure 1** eGate Integrator



Enterprise Manager provides a unified, browser-based framework for managing all aspects of the run-time environment, as well as installing and updating all ICAN Suite components. Enterprise Designer provides a unified, graphical development environment for integrating systems and developing composite applications using Web Services.

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 ICAN 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.

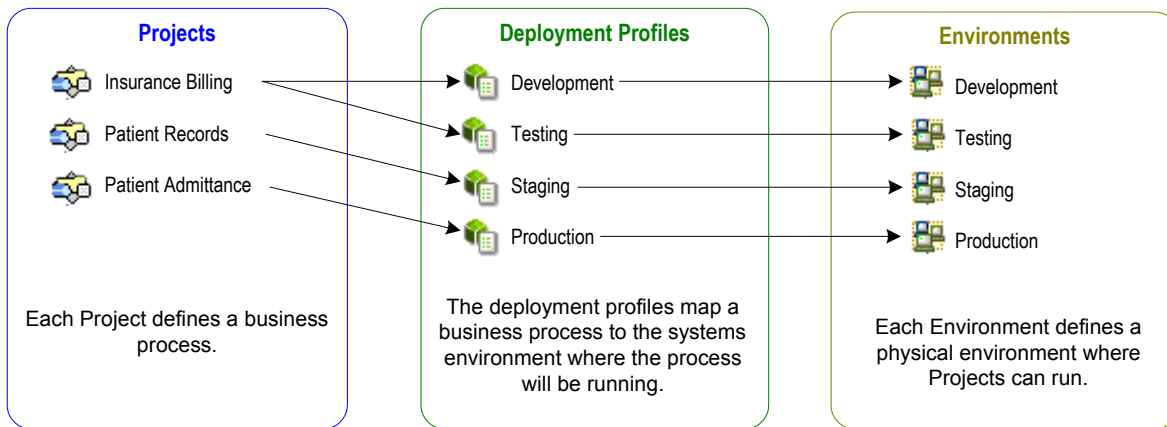
Projects are created using tools contained within Enterprise Designer and, once deployed, can be run and monitored using Enterprise Manager. Projects can also be set up to be run from the business process level using the SeeBeyond eInsight Business Process Manager, if that product is also installed.

Projects are run within individual sets of system definitions, referred to as Logical Hosts. These 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 realms. 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 realm without having to make changes to the other realm.

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 2** 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, implementing an integration Project using eGate Integrator includes the following steps:

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

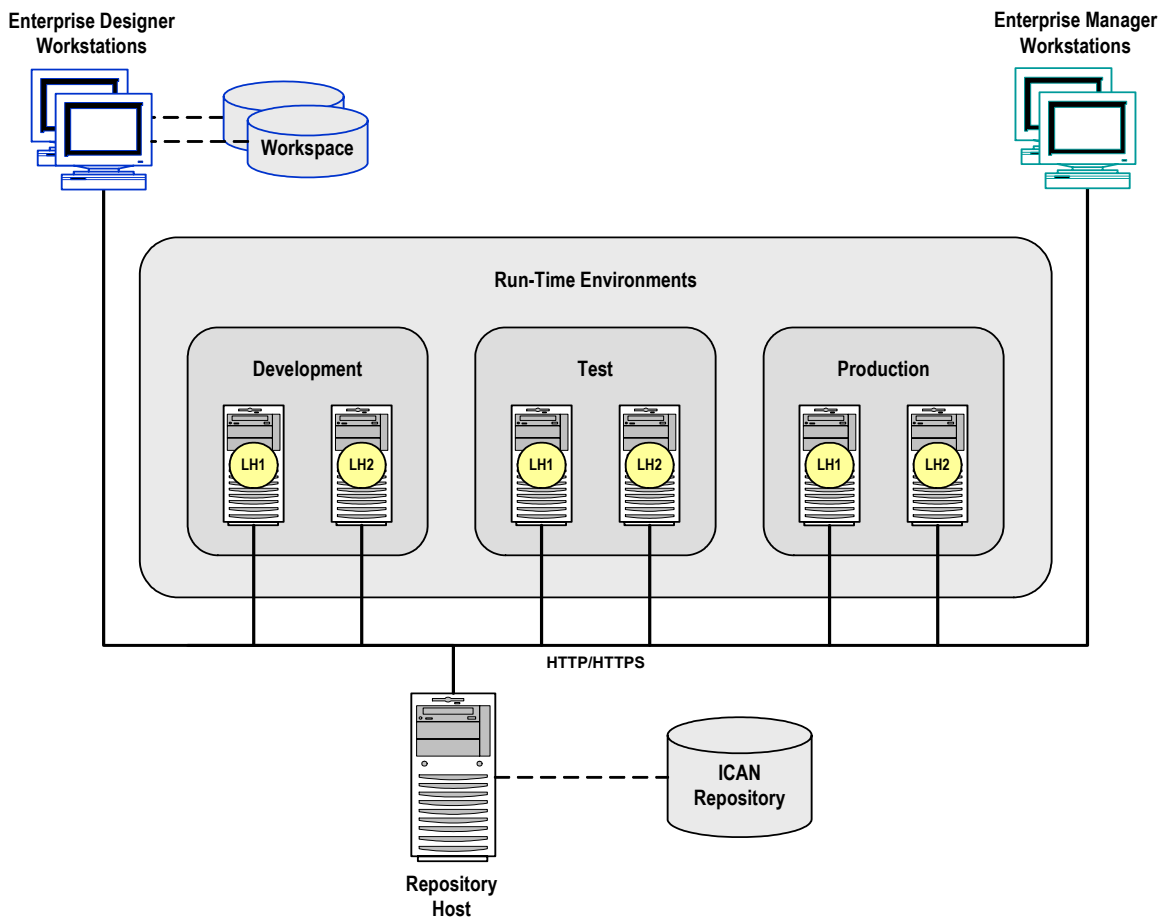
These implementation steps are all accomplished using Enterprise Designer, which is introduced in [Enterprise Designer](#) on page 28 and developed further in subsequent chapters.



## 2.3 System Architecture

SeeBeyond's eGate Integrator employs a versatile architecture that is ideally suited to 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 3 shows an example system implementation that is highly distributed.

**Figure 3** Distributed eGate Integrator System



**Note:** In this scenario it is assumed that all instances of eGate Integrator are of the same release.

## 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 run time 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 3, a single Repository serves the entire enterprise. This common Repository is used for development, testing, and production purposes. Communication between the Repository and other ICAN components can be configured to use either HTTP or HTTPS. The Enterprise Designer and Enterprise Manager clients can communicate with the Repository through a firewall. The Repository makes Web Services available via a UDDI registry.

## 2.3.2 Run-Time Environments

An ICAN Environment represents the total 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 system definitions. Each definition must include one or more **integration servers** such as the SeeBeyond Integration Server, which are the engines that run eInsight Enterprise Service Bus services 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). Each collection of integration servers, message servers, and additional software modules comprise what is known as a *Logical Host*.

- **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 3, the production environment is split across two hardware platforms, each supporting 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 ICAN 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 3 to run on separate systems; all could run on a single system, provided that resources (CPU, memory and disk) are sufficient to support concurrent usage.

## 2.4 User Interfaces

eInsight Enterprise Service Bus uses two basic graphical user interfaces (GUIs), each of which addresses a different set of users. Enterprise Manager is an interface used by the entire ICAN Suite, the primary users of which are system administrators. Enterprise Designer is used by personnel who are involved in defining a software system for integrating the various enterprise applications using eInsight Enterprise Service Bus and eInsight Business Process Manager.

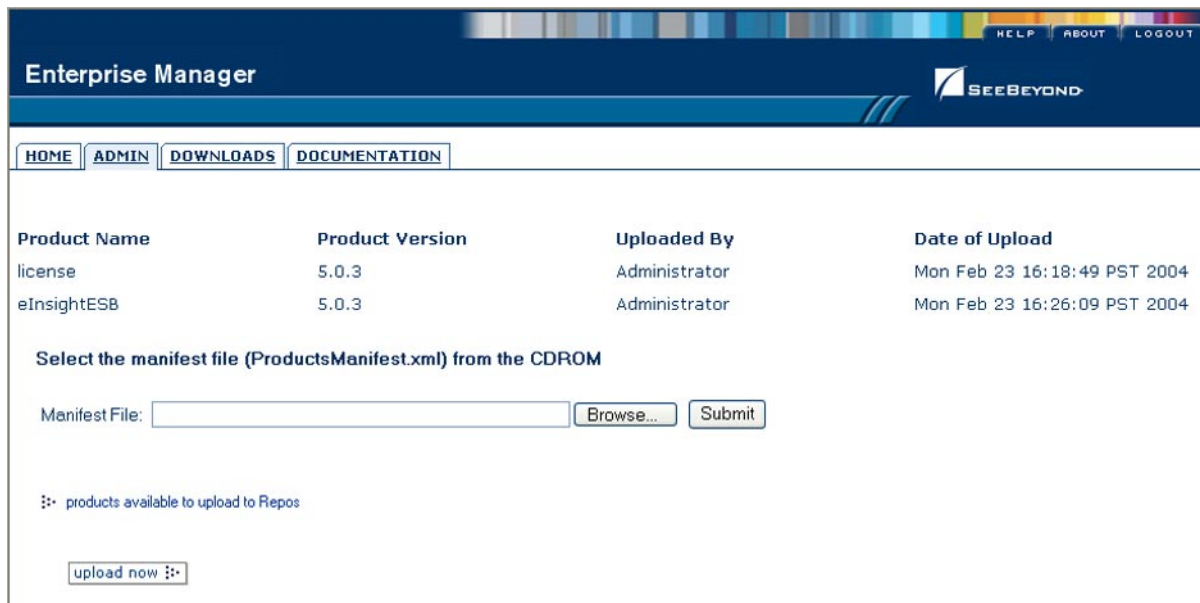
### 2.4.1 Enterprise Manager

Enterprise Manager is a Web-based application that works within Microsoft Internet Explorer. It is used throughout the SeeBeyond ICAN Suite for:

- Installing and updating ICAN Suite products
- Accessing ICAN Suite product documentation
- Managing and monitoring runtime components

The Enterprise Manager is described in [Enterprise Manager on page 30](#). Figure 4 shows the Enterprise Manager **Admin** page, used in product installation.

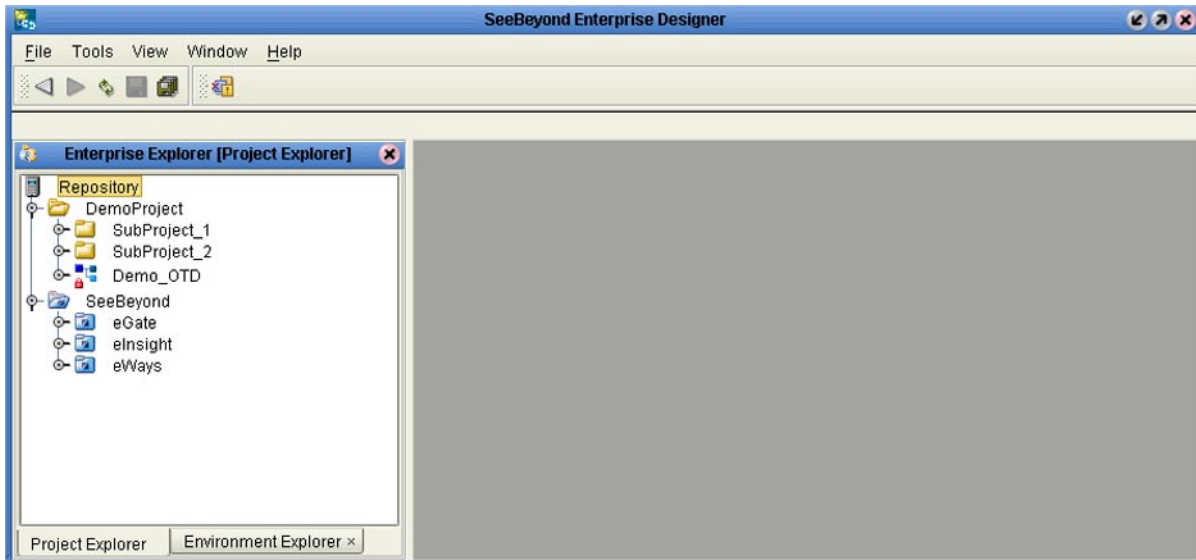
**Figure 4** SeeBeyond Enterprise Manager



## 2.4.2 Enterprise Designer

The SeeBeyond Enterprise Designer is used to create and configure the logical components and physical resources of an ICAN Project. Through this GUI (see Figure 5), you can develop Projects to process and route data through an eInsight Enterprise Service Bus system. Enterprise Designer is also used by other components of the ICAN Suite.

**Figure 5** Enterprise Designer

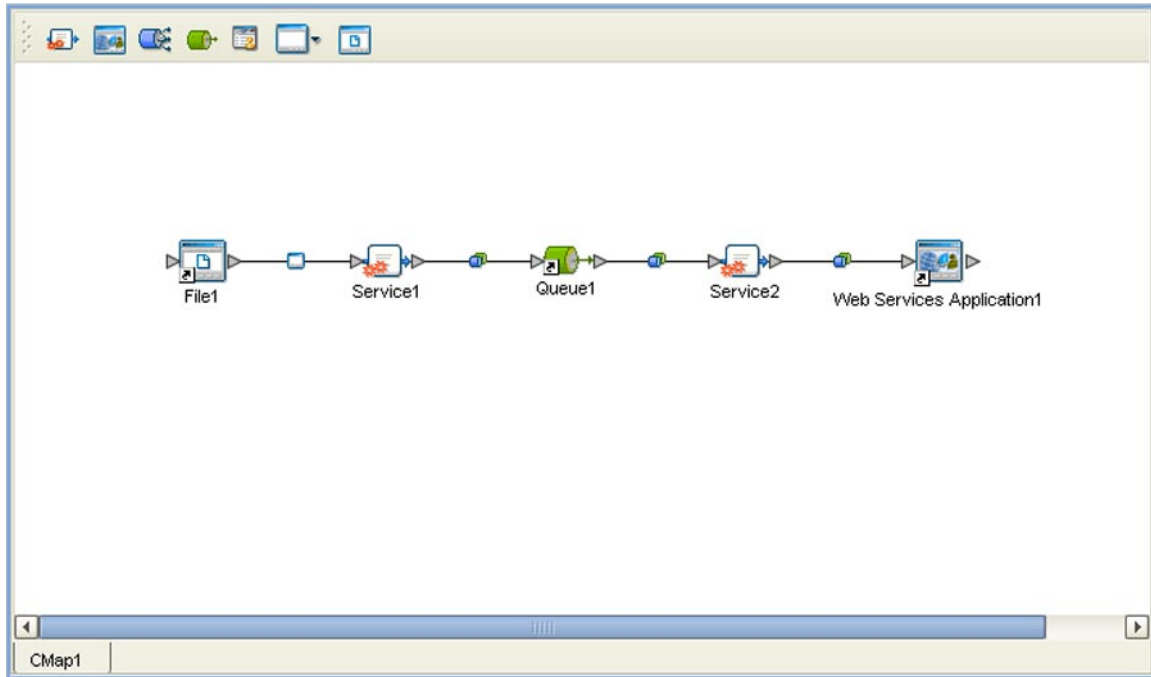


The major features of the Enterprise Designer are the Enterprise Explorer on the left, and an editor panel on the right—which is initially blank. The Enterprise Explorer follows the familiar Windows Explorer format, displaying a tree structure. The editor panel displays a variety of editors, depending upon what component is selected in the Enterprise Explorer. These editors include, for example:

- Connectivity Map Editor
- OTD Editor
- Environment Editor
- Deployment Editor

The Connectivity Map Editor (see Figure 6) provides a graphic example of one of these editors, in which logical components of a Project can be created and connected. eInsight Enterprise Service Bus uses Connectivity Maps to intuitively configure the end-to-end flow of messages within an integration. The integration developer can drag and drop the various components onto the Connectivity Map canvas and link them together to specify message flow. The features and usage of the Connectivity Map Editor are described in [ICAN Projects](#) on page 73.

**Figure 6** Connectivity Map Editor



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 36](#).

# Enterprise Manager

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

## What's in This Chapter

- **Overview** on page 30
- **Starting Enterprise Manager** on page 31
- **The Enterprise Manager Interface** on page 32
- **The ICAN Monitor** on page 35

---

## 3.1 Overview

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

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

### 3.1.1 Installing and Updating eInsight Enterprise Service Bus

eInsight Enterprise Service Bus 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 eInsight Enterprise Service Bus components, see the *SeeBeyond ICAN Suite Installation Guide*.

### 3.1.2 Monitoring and Managing eInsight Enterprise Service Bus

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

- **The ICAN Monitor** on page 35 describes the basic features of the ICAN Monitor interface. ICAN Monitor usage for specific tasks is described in the *eGate Integrator System Administration Guide*.

## 3.2 Starting Enterprise Manager

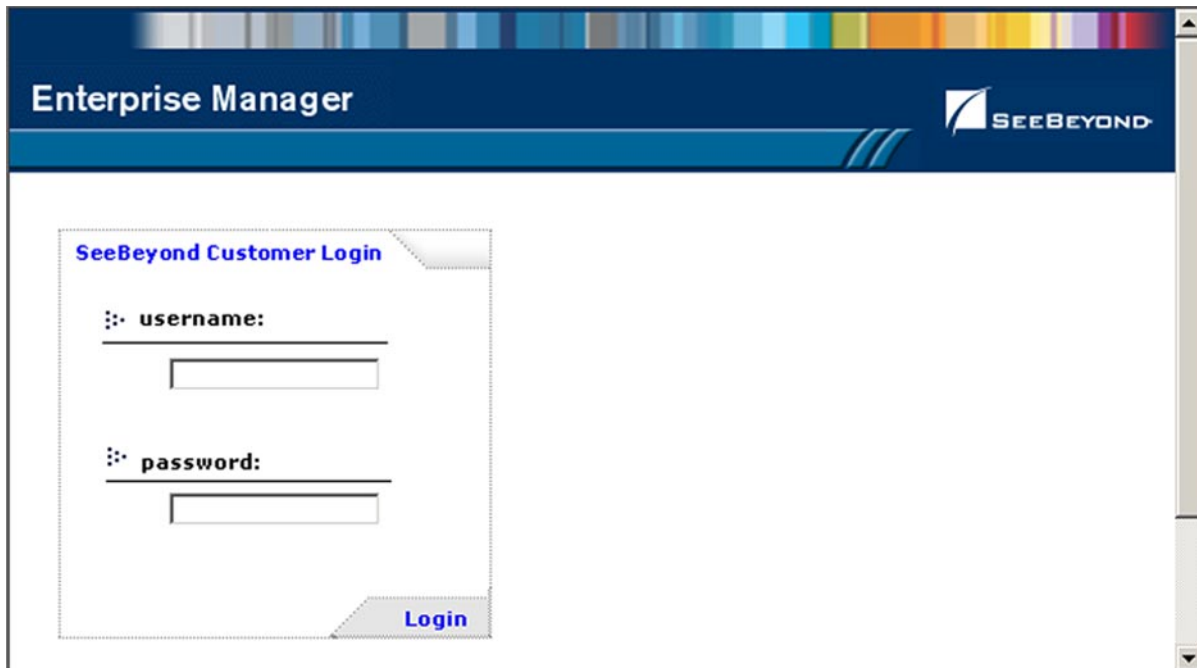
To start the Enterprise Manager

- 1 Launch Internet Explorer.
- 2 Enter **http://hostname:portnumber** in the **Address** box to display the SeeBeyond Customer Login window shown in Figure 7.

**Note:** The *hostname* is the fully-qualified, network-addressable 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 7** Enterprise Manager Login



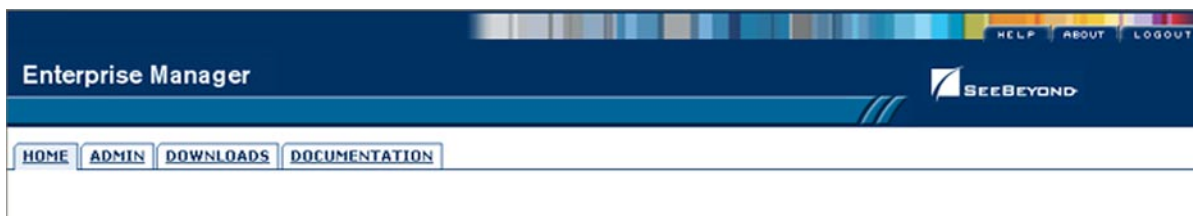
The screenshot shows a web browser window displaying the SeeBeyond Customer Login page. The page has a dark blue header with the text 'Enterprise Manager' on the left and the SeeBeyond logo on the right. Below the header is a white content area. In the center of the content area is a login form titled 'SeeBeyond Customer Login'. The form contains two input fields: 'username:' and 'password:', each with a small icon to its left. Below the password field is a 'Login' button.

- 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 8).

**Figure 8** Enterprise Manager GUI



The Enterprise Manager is organized into four pages, as described in the following table. Each page is accessed by clicking the appropriate tab.

**Table 2** Enterprise Manager - Pages

Page	Function
Home	The Home page is used for accessing the ICAN Monitor, which is the main subject of this chapter. See <a href="#">Home</a> on page 33.
Admin	The Administration page is used for installing and updating ICAN components. See the <i>SeeBeyond ICAN Suite Installation Guide</i> for information.
Downloads	The Downloads page is used in installing and updating ICAN components. See the <i>SeeBeyond ICAN Suite Installation Guide</i> for information.
Documentation	The Documentation page is used for accessing ICAN Suite documentation and sample Projects. See <a href="#">Documentation</a> on page 34, and the following <i>Note</i> .

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

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 - Control 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 not present on the Documentation 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** page (see Figure 9) contains a link to the ICAN Monitor. Click the **Monitor** icon to launch the ICAN Monitor (see [The ICAN Monitor](#) on page 35).

**Figure 9** ICAN Monitor Launch Icon

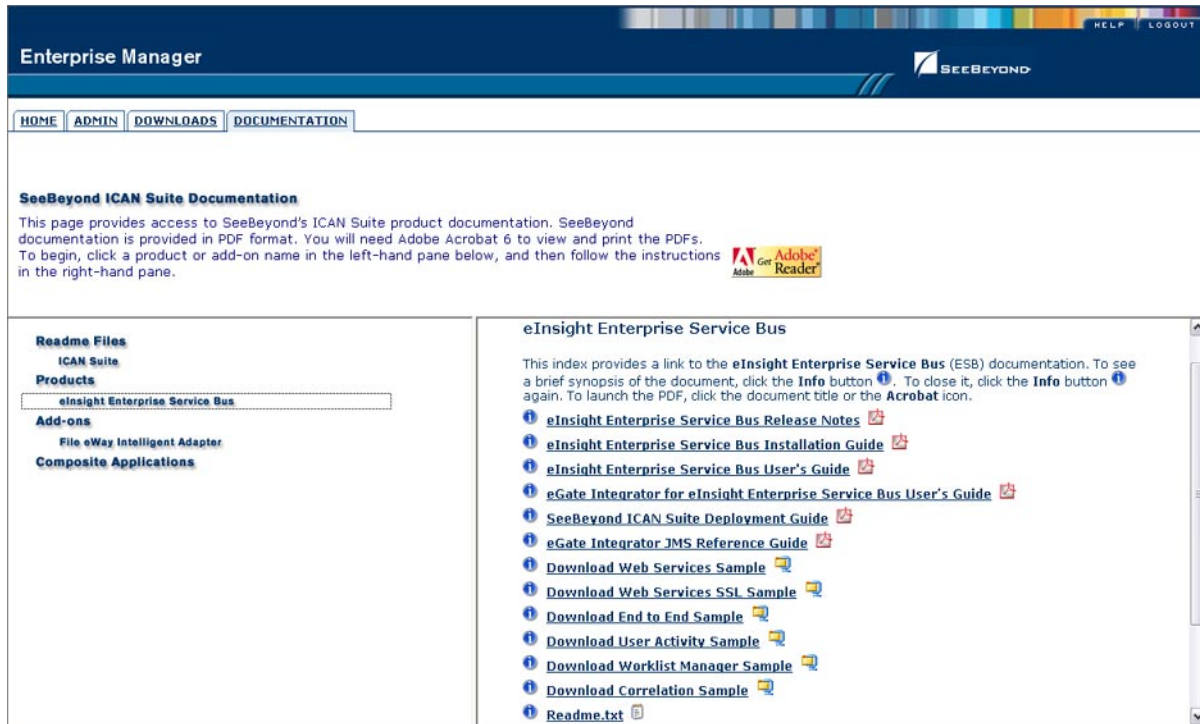


**Note:** *If connection problems are encountered, close all Internet Explorer windows and retry.*

### 3.3.2 Documentation

The **Documentation** page (see Figure 10) contains links to the uploaded versions of the SeeBeyond ICAN documentation in Adobe Acrobat (PDF) format, and also any sample Project files (in ZIP format). Shown is the current set for eInsight Enterprise Service Bus.

**Figure 10** Documentation Page



The provided documentation is organized into the major categories listed in Table 4.

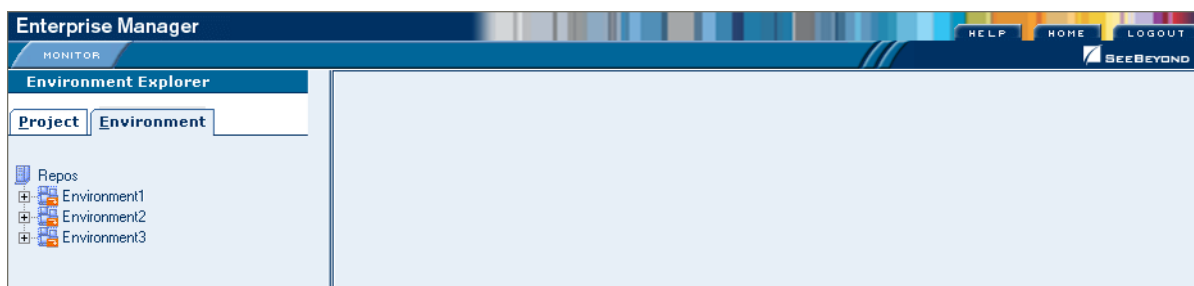
**Table 4** Document Categories

Category	Contents
Readme Files	Includes information regarding the latest operating system and hardware requirements, cautions and caveats regarding known issues, and supplementary information arising after the publication of other documentation.
Products	Documentation regarding ICAN core products, such as eInsight Enterprise Service Bus. Also includes any installed example Project files.
Add-ons	Documentation regarding optional, ancillary products such as eWays and OTD Libraries.
Composite Applications	Documentation regarding ICAN Composite Applications, such as Corporate Actions or X12 Protocol Manager.

## 3.4 The ICAN Monitor

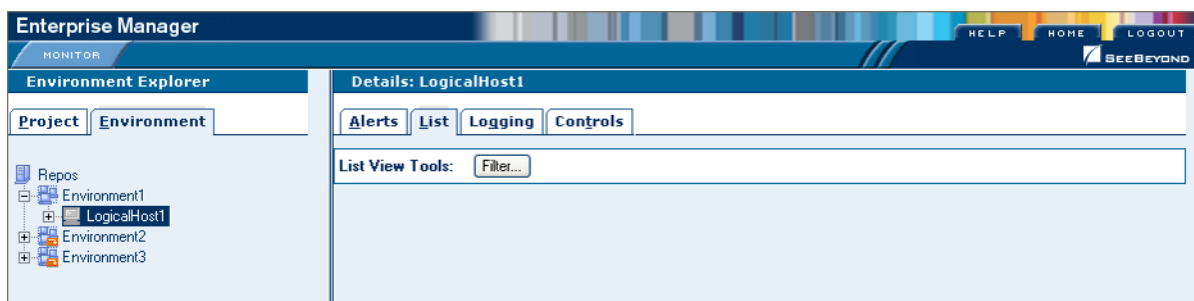
The ICAN Monitor has structure similar to that of the Enterprise Designer, with an Explorer panel on the left and a Details panel on the right. Initially, the Details panel is blank as shown in Figure 11.

**Figure 11** ICAN Monitor Interface - Initial



Like the Enterprise Manager itself, the ICAN Monitor's **Details** area is organized into sections represented by tabs (see Table 5). 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 12.

**Figure 12** ICAN 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.

**Table 5** ICAN Monitor Interface - Details Tabs

Tab	Function
Alerts	Displays all alerts for the component selected in the Explorer.
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.
Controls	Displays controls that allow an Administrator to intervene in the run-time process and perform tasks such as starting and stopping components.

**Note:** See the *eGate Integrator System Administration Guide* for detailed information regarding Monitor usage.

# Enterprise Designer

This chapter presents an overview of the major features of the Enterprise Designer.

## What's in This Chapter

- **Overview** on page 36
- **Using a Proxy Server** on page 38
- **Starting Enterprise Designer** on page 40
- **Interface Features** on page 42
- **Enterprise Explorer** on page 47
- **Enterprise Designer Editors** on page 49
- **Additional Tools and Features** on page 53

---

## 4.1 Overview

### 4.1.1 User Interface

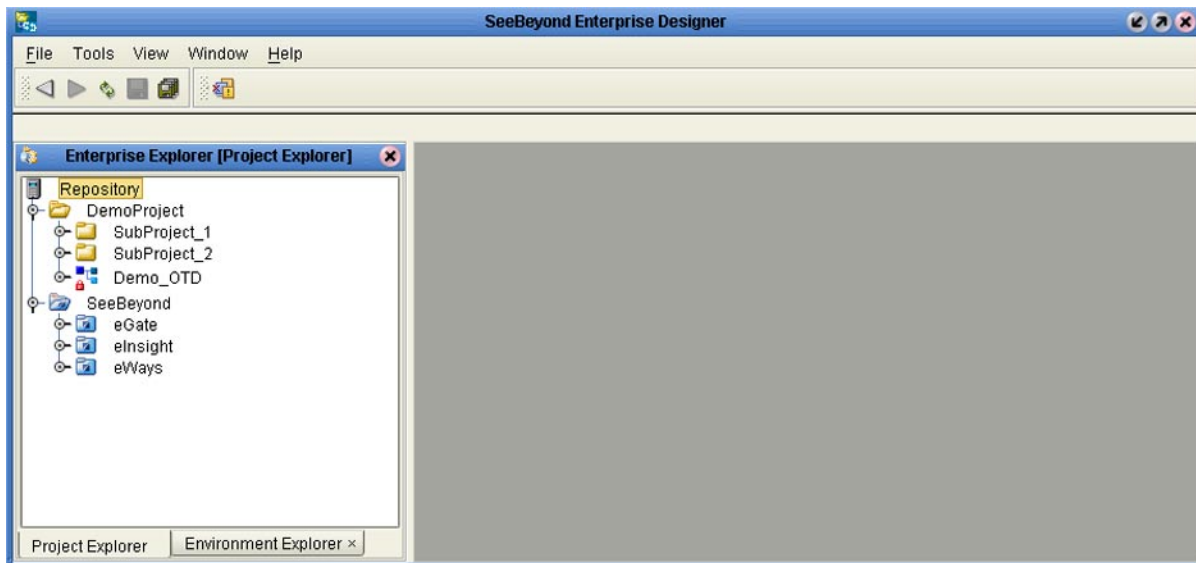
The Enterprise Designer graphical user interface is used to create and configure the logical components and physical resources of an ICAN Project. Enterprise Designer contains the customary graphical interface features, which are described in the following sections of this chapter.

- **Menus** on page 42 describes the options contained in the individual menus.
- **Toolbar** on page 45 describes the functionality of the toolbar icons.
- **Browser Buttons** on page 46 describes the browser buttons that appear throughout the Enterprise Designer, in various wizards and dialog boxes.

The major features of Enterprise Designer are the Enterprise Explorer on the left, and an editor panel on the right—which is initially blank (see Figure 13). 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 47
- **Environment Explorer on page 48**

**Figure 13** Enterprise Designer User Interface



## 4.1.2 Editors

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 49
- **OTD Editor** on page 50
- **Environment Editor** on page 51
- **Deployment Editor** on page 52

## 4.1.3 Analysis and Archiving Tools

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

- **Project/Environment Import** on page 53, which allows you to import a Project that has been created elsewhere.
- **Project/Environment Export** on page 57, which allows you to export a Project to an external file so that it may be used elsewhere.
- **Impact Analyzer** on page 62, which helps you visualize how a change to one part of a Project would affect the rest of the Project.
- **Version Control** on page 64, which allows you to maintain multiple versions of Project components.

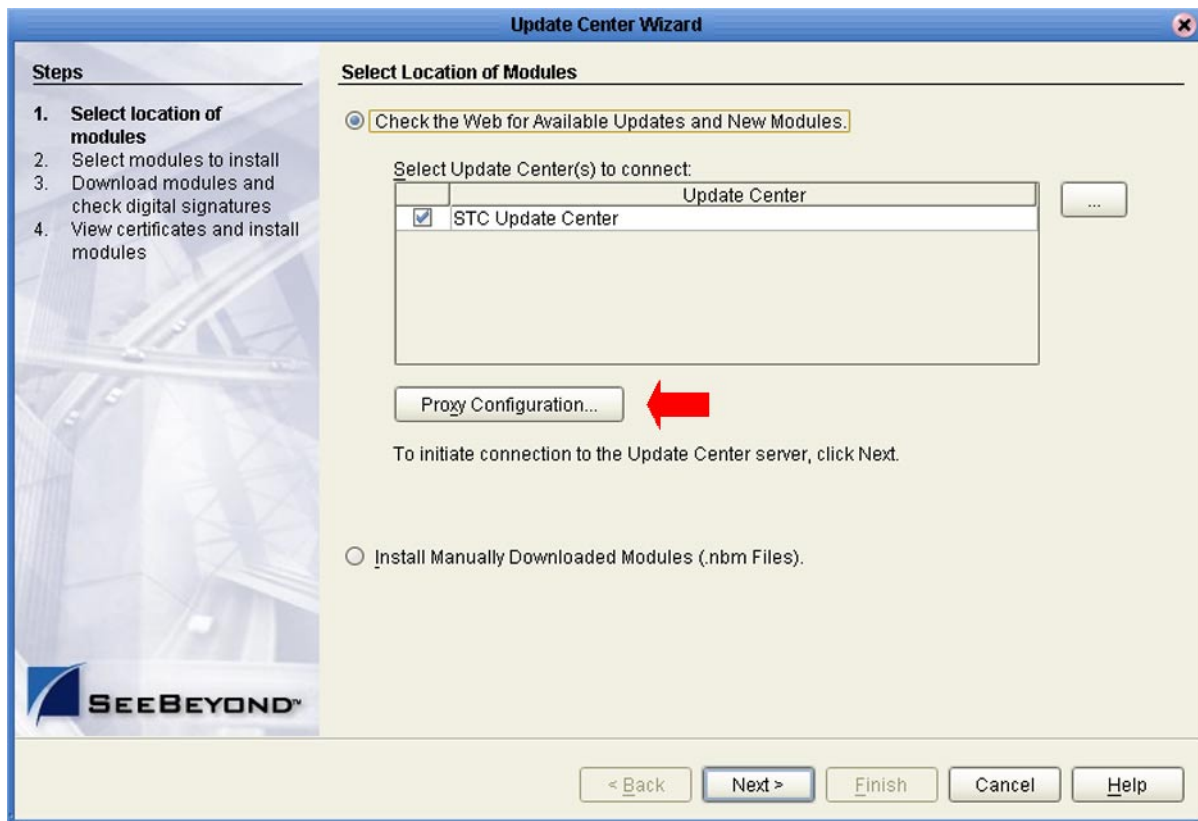
## 4.2 Using a Proxy Server

The following procedure allows Enterprise Designer to reference a WSDL file using a URL that points outside your firewall. See the *SeeBeyond ICAN Suite Installation Guide* for additional information.

To configure a proxy server

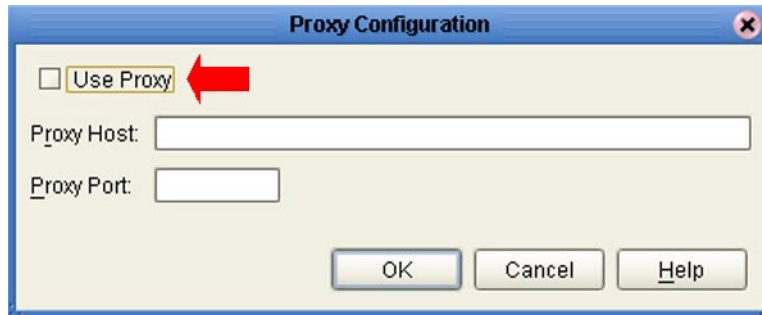
- 1 From Enterprise Designer's **Tools** menu, select **Update Center** to display the Update Center Wizard (see Figure 14).

**Figure 14** Update Center Wizard



- 2 Click the **Proxy Configuration** button (see Figure 14) to display the Proxy Configuration dialog box shown in Figure 15.

**Figure 15** Proxy Configuration Dialog Box



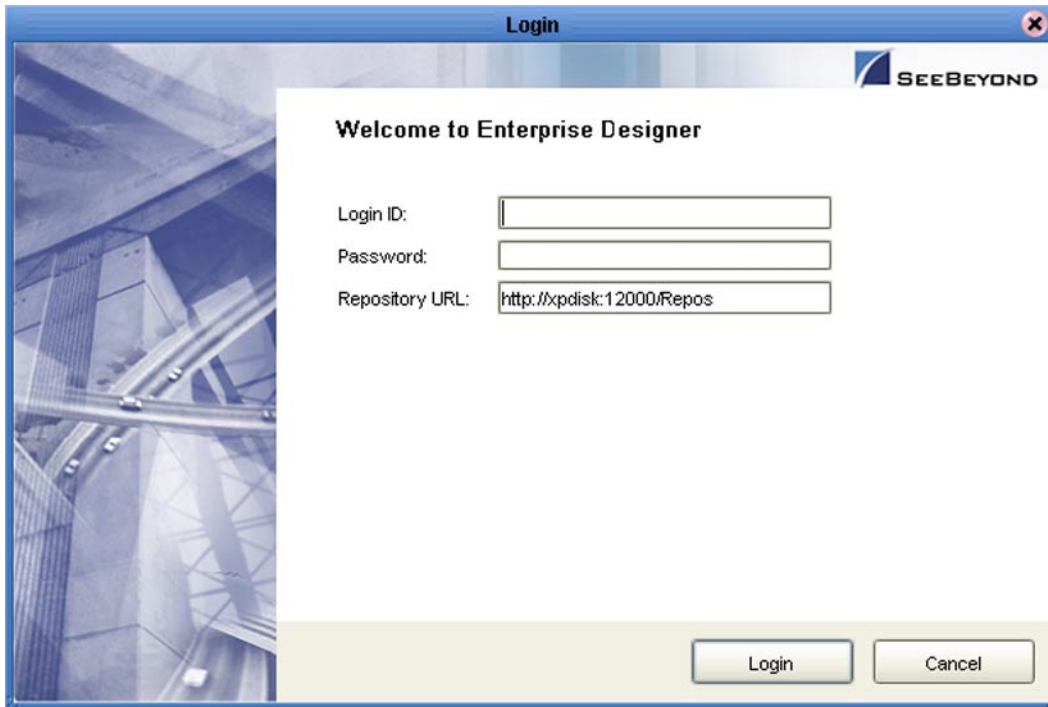
- 3 Check the **Use Proxy** box, and fill in the fully-qualified name of the proxy host and port for the proxy server.
- 4 Click **OK** to finish.

## 4.3 Starting Enterprise Designer

### To start the Enterprise Designer on a Windows Platform

- 1 Run the batch file `ICAN-root\edesigner\bin\runed.bat` to display the *Login* dialog box shown in Figure 16 (placing a shortcut on your desktop streamlines this procedure).

**Figure 16** 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 The URL for the Repository should be displayed in the *Repository URL* text box. If it is incorrect, edit the URL before proceeding. See the *SeeBeyond ICAN Suite Installation Guide* for details.
- 5 Click **Login** to complete the login process and display the Enterprise Designer GUI shown in Figure 13. A progress monitor will appear while the process is running.

### To start the Enterprise Designer on a UNIX Platform

- 1 Run the script `ICAN-root/edesigner/bin/runed.sh` to display the *Login* dialog box shown in Figure 16.
- 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 The URL for the Repository should be displayed in the *Repository URL* text box. If it is incorrect, edit the URL before proceeding. See the *SeeBeyond ICAN Suite Installation Guide* for details.
- 5 Click **Login** to complete the login process and display the Enterprise Designer GUI shown in Figure 13.

## 4.4 Interface Features

### 4.4.1 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
Save	Saves changes to the selected objects (to the local workspace only).
Save All	Saves changes to all objects currently open in the editor (to the local workspace only).
Exit	Closes the Enterprise Designer.

#### Tools Menu

**Table 7** Tools Menu Options

Option	Function
Impact Analyzer	Displays a dialog box in which you can view how one component of a Project impacts other components. See <a href="#">Impact Analyzer</a> on page 62.
Options	Displays the Options Setup dialog box, in which you can specify selected options such as heap sizes and language extensions. See <a href="#">Options Setup</a> on page 44.
Update Center	Displays a series of dialog boxes in which you can check for program updates. See the <i>SeeBeyond ICAN Suite Installation Guide</i> .

#### View Menu

**Table 8** View Menu Options

Option	Function
Environment Explorer	Activates the <b>Environment Explorer</b> tab on the Enterprise Explorer. See <a href="#">Environment Explorer on page 48</a> .
Project Explorer	Activates the <b>Project Explorer</b> tab on the Enterprise Explorer. See <a href="#">Project Explorer on page 47</a> .

## 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 Editor canvas.
Restore All	Returns minimized windows to their original position on the Editor canvas.
Close All	Closes all open windows.

## Help Menu

**Table 10** Help Menu Options

Option	Function
About Enterprise Designer	Displays an information box giving the version number, copyright information, and ICAN Repository connection information.
Contents	Displays the online help for all installed components of the ICAN Suite that operate within Enterprise Designer.
Help Sets	For future use (currently duplicates the Contents option).

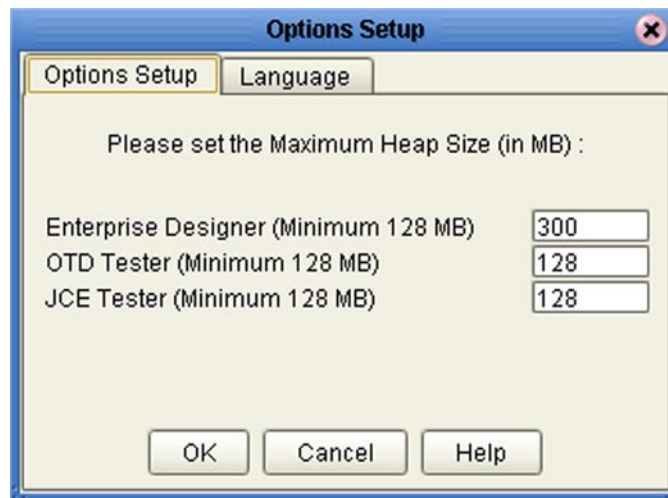
## Options Setup

### Options Setup

The *Options Setup* tab (see Figure 17) allows you to increase the heap size of Enterprise Designer itself and the OTD Tester (the JCE Tester is not included with eInsight ESB) to accommodate large file sizes, if necessary.

**Note:** *If you encounter an out-of-memory error, try increasing the heap size in increments of 50Mb.*

**Figure 17** Options Setup - Heap Size Dialog



### Language







The *Language* tab allows you to control the use of extended language options in International versions of eGate Integrator (not used in eInsight ESB).

**Figure 18** Options Setup - Language Dialog



## 4.4.2 Toolbar






**Table 11** Enterprise Designer Toolbar Icons

Icon	Command	Function
	Step Back	Steps back in your usage history for the current session.
	Step Forward	Steps forward in your usage history for the current session.
	Refresh All from Repository	Refreshes the Project Explorer and Environment Explorer to display the current contents of the Repository. (You are prompted to save any changes before the refresh occurs.) Open editors are not refreshed.
	Save	Saves changes made to the selected Project to the local workspace only—the Repository is <i>not</i> updated. This icon is inactive if no changes have been made.
	Save All	Saves changes made to all open Projects to the local workspace only—the Repository is <i>not</i> updated. This icon is inactive if no changes have been made.
	Impact Analyzer	Displays the <i>Impact Analyzer</i> dialog box, which allows you to view how one component of a Project impacts other components.

### 4.4.3 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	Command	Function
	Up One Level	Returns you to the parent folder or directory.
	Home	Returns you to the root folder or directory, or—depending upon the context—the default object.
	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.).

## 4.5 Enterprise Explorer

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

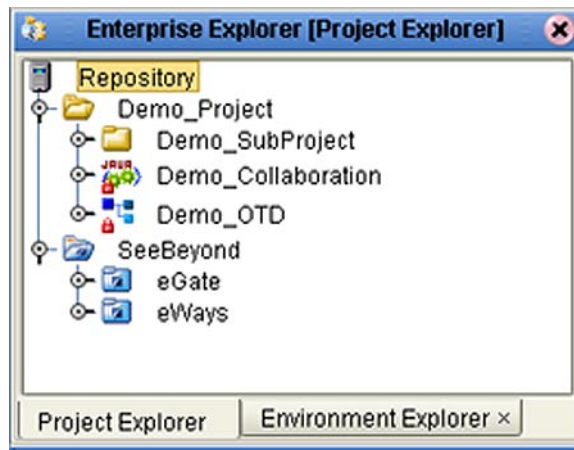
- **Project Explorer** on page 47 deals with logical components.
- **Environment Explorer on page 48** deals with physical resources, including the Logical Host and Integration Server.

**Note:** *The Project and Environment trees are initially loaded only to the Project or Environment level. The contents of a Project or Environment are loaded when you expand the particular node. This causes a slight delay when you expand the node, but eliminates a potentially-significant delay when you open Enterprise Designer, due to the large size of some OTD libraries.*

### 4.5.1 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 19.

**Figure 19** Enterprise Explorer: Project Explorer View

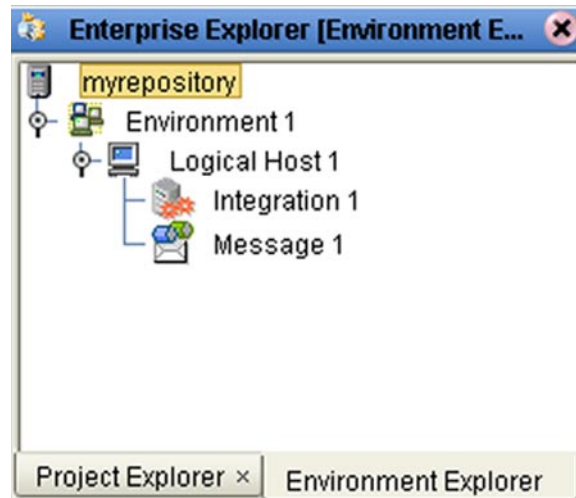


Details of the features and usage of the Project Explorer are found in **ICAN Projects** on page 73.

## 4.5.2 Environment Explorer

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

**Figure 20** Enterprise Explorer: Environment Explorer View



Details of the features and usage of the Environment Explorer are found in [Environments](#) on page 123.



## 4.6 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.

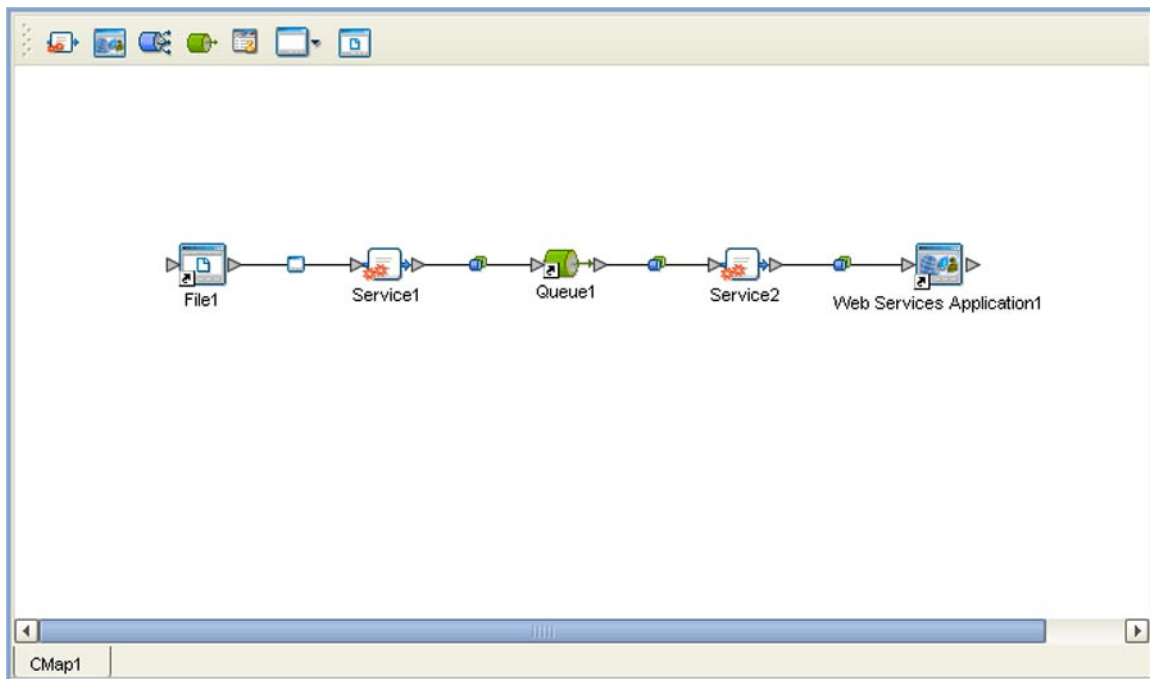
*Note:* See the *eInsight Enterprise Service Bus User's Guide* for a detailed explanation of the steps involved in setting up an eInsight ESB Project.

### 4.6.1 Connectivity Map Editor

A Connectivity Map is a graphical representation of your Project, containing the various logical components comprising the Project and the links between them. The Connectivity Map Editor, shown in Figure 21, 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.

See [Using the Connectivity Map Editor](#) on page 86 for detailed information.

**Figure 21** Connectivity Map Editor

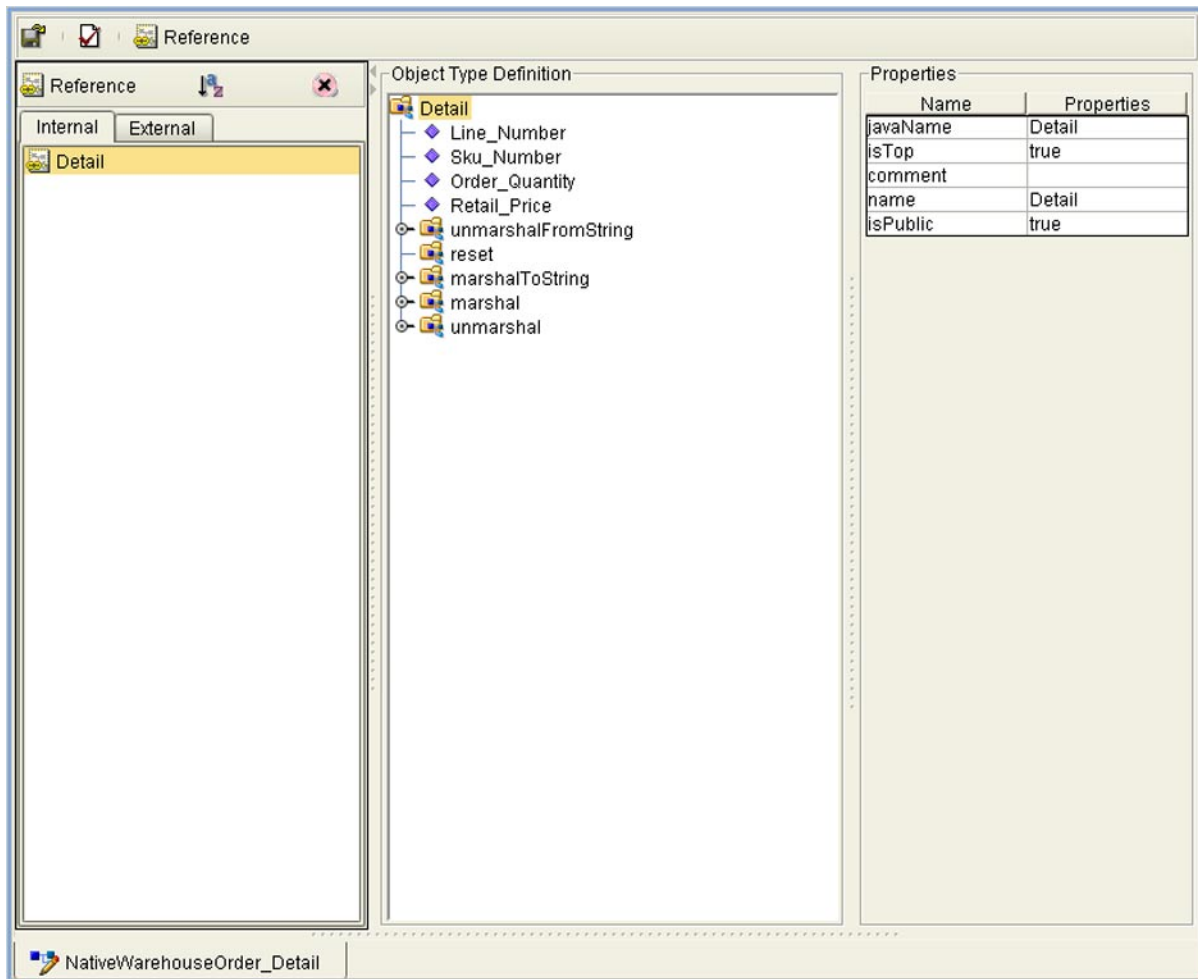


## 4.6.2 OTD Editor

The OTD Editor window, as shown in Figure 22, displays the source files used to create the Object Type Definitions (OTDs) to use with a Project. You use an OTD wizard tool to create OTD files and add them to the **Project Explorer** tab.

See [OTD Editor](#) on page 100 for detailed information.

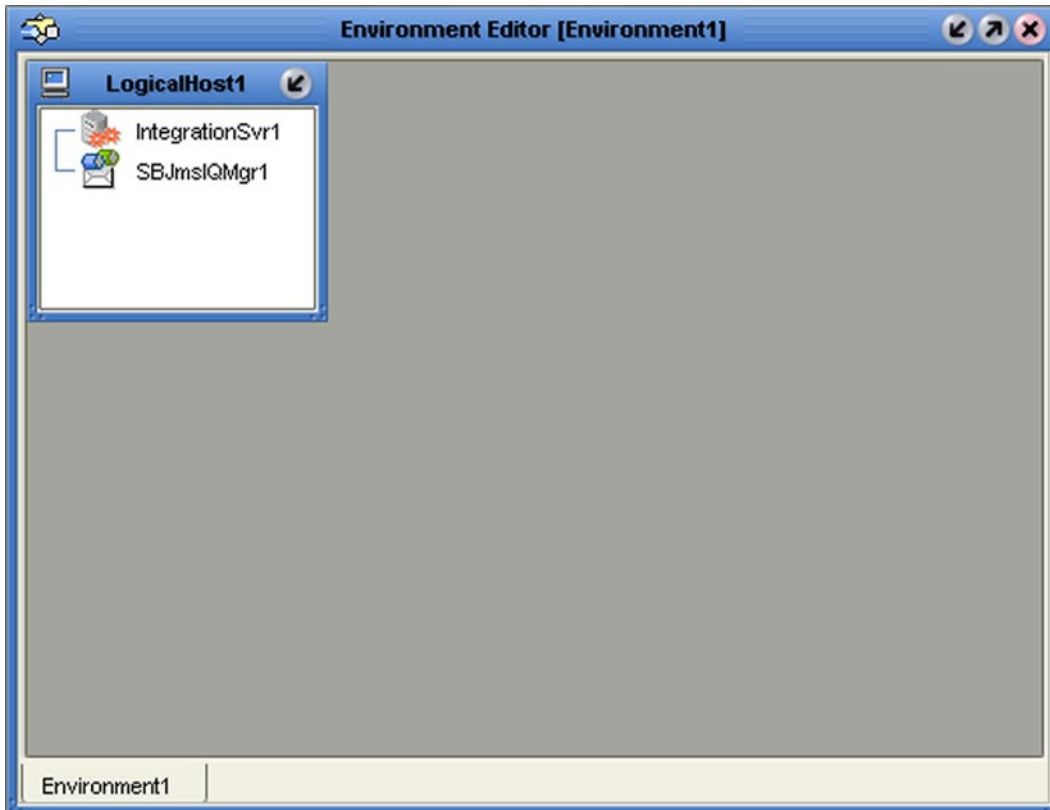
**Figure 22** OTD Editor



### 4.6.3 Environment Editor

The Environment Editor provides a canvas in which you can create and customize a run-time Environment. Here you can see the various components (Logical Hosts, servers, and external systems) included in the selected Environment. An Environment containing a populated Logical Host is shown in Figure 23.

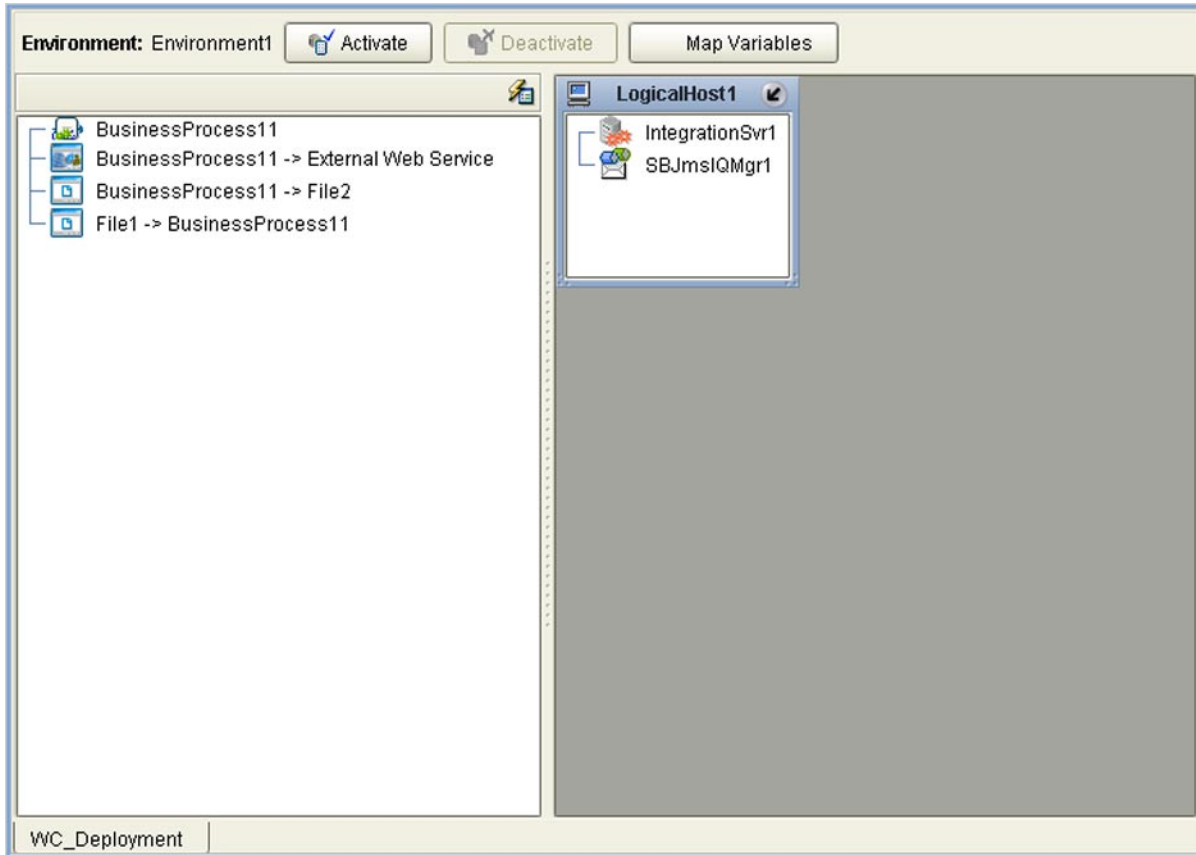
**Figure 23** Environment Editor



## 4.6.4 Deployment Editor

The Deployment Editor, as shown in Figure 24, contains information about how Project components will be deployed in an Environment. See [Deployment Editor](#) on page 175 for detailed information

**Figure 24** Deployment Editor



## 4.7 Additional Tools and Features

### 4.7.1 Project/Environment Import

The import function allows you to import an ICAN Project or Environment file using the Enterprise Designer. Both follow essentially the same procedure.

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

When importing a Project, note that:

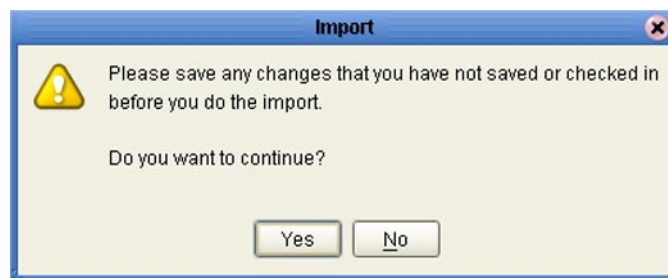
- Existing Projects are not affected by the imported Project.
- During import, if another Project having the same name exists in the target Repository, you will receive an error message and the existing file will not be overwritten.
- If you have not installed all of the necessary products (such as eWays) that a Project requires, you will not be able to import that Project and will get an error message.
- References are validated during import.
- Project deployment objects are not imported (they have references to both Project and Environment elements that are not required at the Project level).

### Importing a Project Using Enterprise Designer

To import a Project using Enterprise Designer

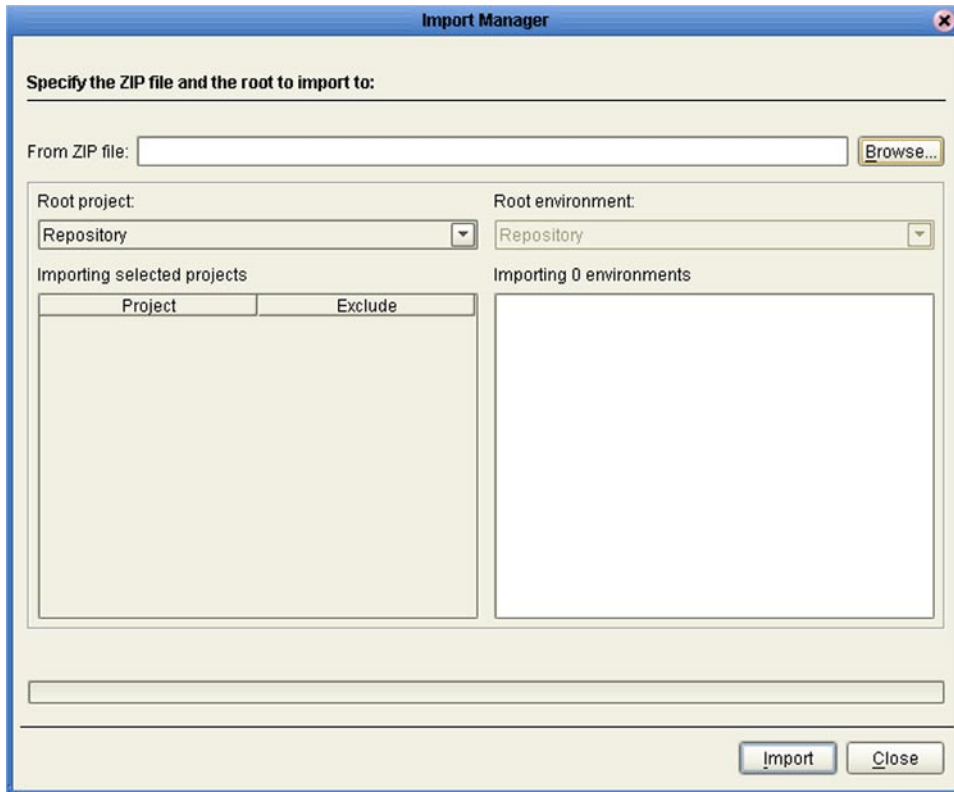
- 1 From the Repository context menu (for Projects) or the Project context menu (for Sub-Projects), select **Import**.
- 2 The message box shown in Figure 25 appears, prompting you to save your changes.

**Figure 25** Import Message Box



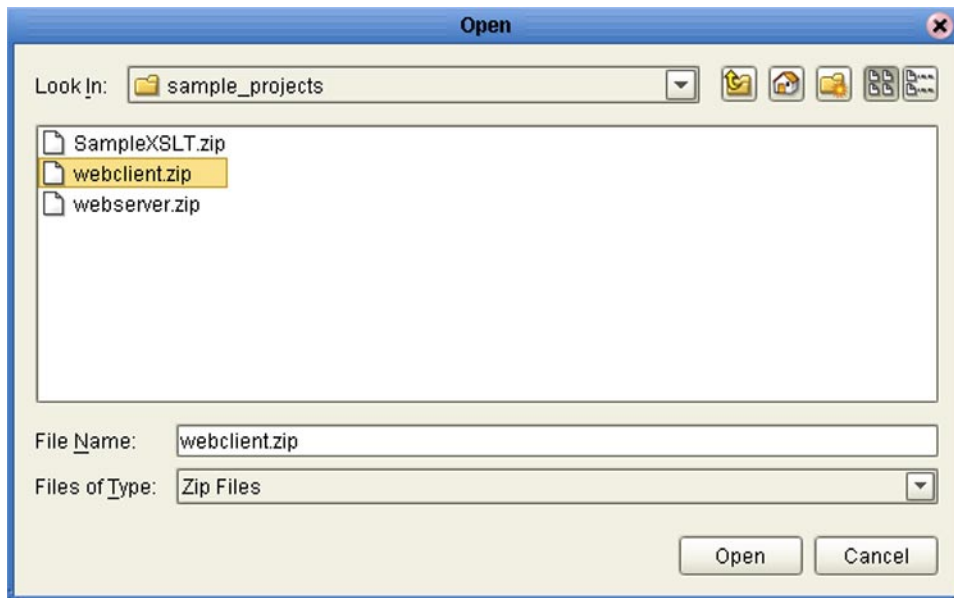
- A If you want to save your changes, but have not already done so, click **No**. Save your changes, and then re-select **Import**, as in step 1.
- B If you have saved any desired changes, click **Yes** to display the dialog box shown in Figure 26.

**Figure 26** Import Manager Dialog Box (1)



- 3 Click the **Browse** button to display the *Open File* dialog box, as shown in Figure 27. If you browse to an Environment file, the *Root environment* field will be enabled.

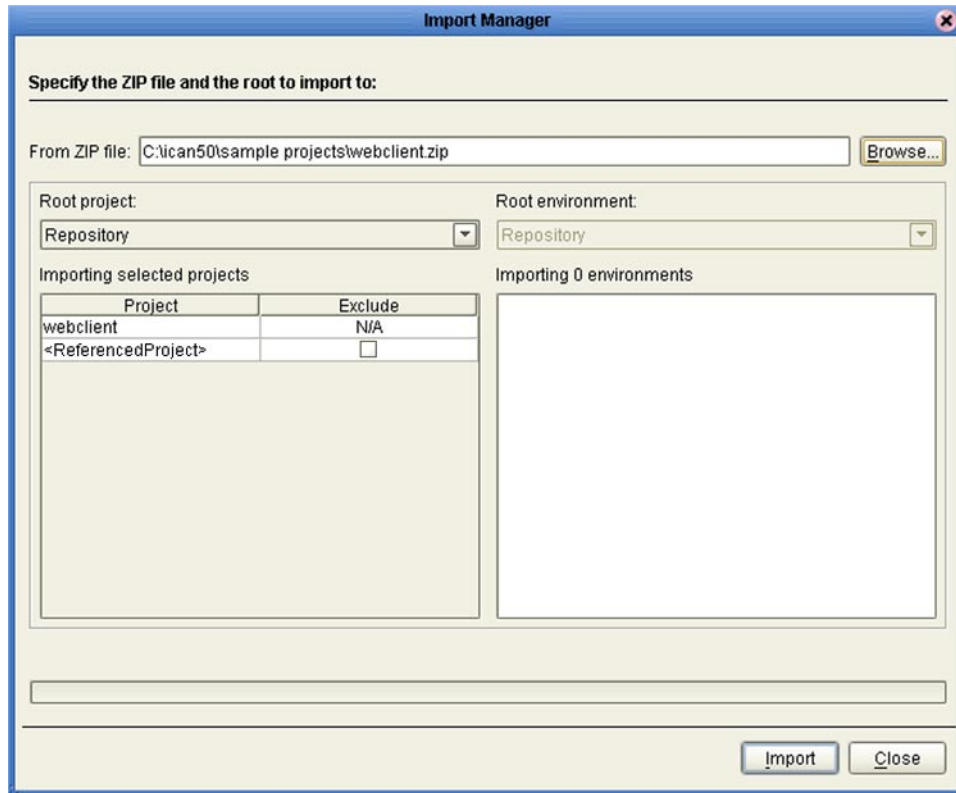
**Figure 27** Open File Dialog Box



- 4 Locate and select the Project or Environment file that you want to import.

- 5 Click **Open** to import the file.
- 6 The Import Manager dialog box appears as shown in Figure 28.

**Figure 28** Import Manager Dialog Box



**Note:** *If the Project you are importing contains references to another Project, and the other Project already resides in your Repository, you have the option of excluding the referenced Project from the import by checking the box that appears in the Exclude column. The references will be retargeted to the Project existing in the Repository.*

- 7 Click **Import** to import the file.
- 8 The Import Status message box shown in Figure 29 appears after the file has been imported successfully.

**Figure 29** Import Status Message Box



- 9 Click **OK** to close the message box.

- 10 When you are finished importing files, click **Close** to close the Import Manager dialog box. The Project Explorer will now automatically be refreshed from the Repository.

## Importing a Project Using the Command Line

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

### Location of script file:

*ICAN-root*\repository\util\importProject.bat (or importProject.sh)

### Command Syntax:

```
importProject username password importfile rootprojectname
```

where:

- ♦ **importfile** is the name and path of the archive file containing the Project or Environment you are importing.
- ♦ **rootprojectname** is the name of the parent Project, under which the imported Project will be a sub-Project. If the Project is not to be imported as a sub-Project, then leave this parameter as an empty string ("").

**Note:** Project names should not include special characters such as a comma (,), single quote ('), or double quote ("). If any of these characters are used, you must escape them when using the importProject command-line tool.

### To import a Project using the import script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To extract a Project contained in the file `c:\project4import.zip` and import it into the Repository, type:

```
importProject username password c:\project4import.zip ""
```

### To import a Project as a sub-Project using the import script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To extract a Project contained in the file `c:\project4import.zip` and import it into the Repository as a sub-Project of **mainProject**, type:

```
importProject username password c:\project4import.zip mainProject
```

### To import an Environment using the import script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To extract an Environment contained in the file `c:\environment4import.zip` and import it into the Repository, type:

```
importProject username password c:\environment4import.zip ""
```

- 3 The Environment, including all Projects deployed to that Environment, will be imported.



## 4.7.2 Project/Environment Export

The export function allows you to export an ICAN Project and/or Environment 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.

### Exporting a Project Using Enterprise Designer

To export a Project or Environment using Enterprise Designer

- 1 From the Project context menu, select **Export** to display the Export Manager dialog box. If you do not have any existing Environments in your Repository, you will see the dialog box shown in Figure 30. If you do, you will see the dialog box shown in Figure 31.

**Figure 30** Export Manager Dialog Box (1a)

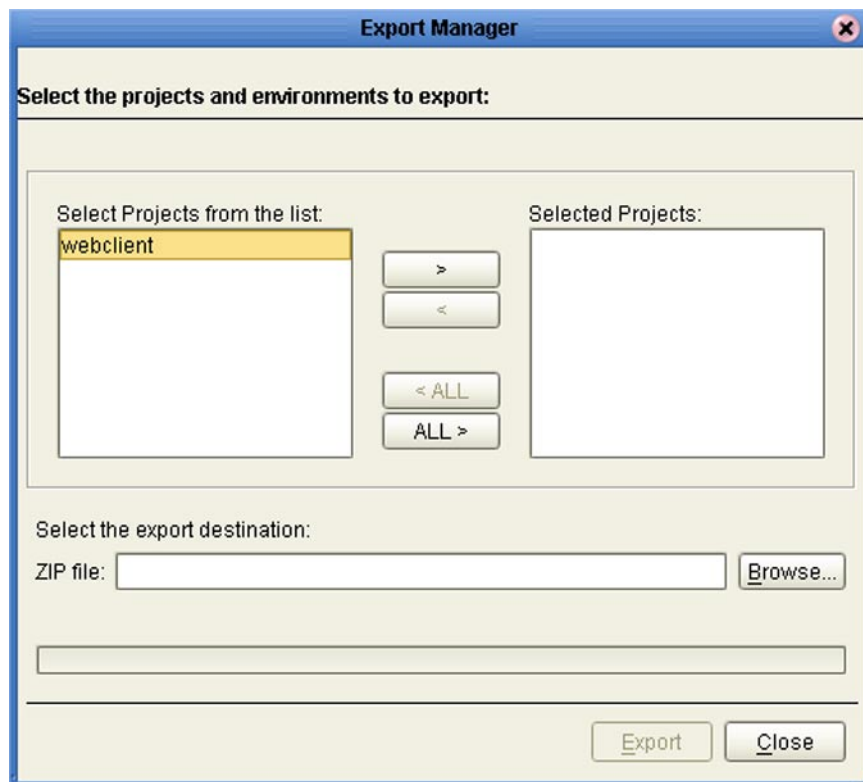
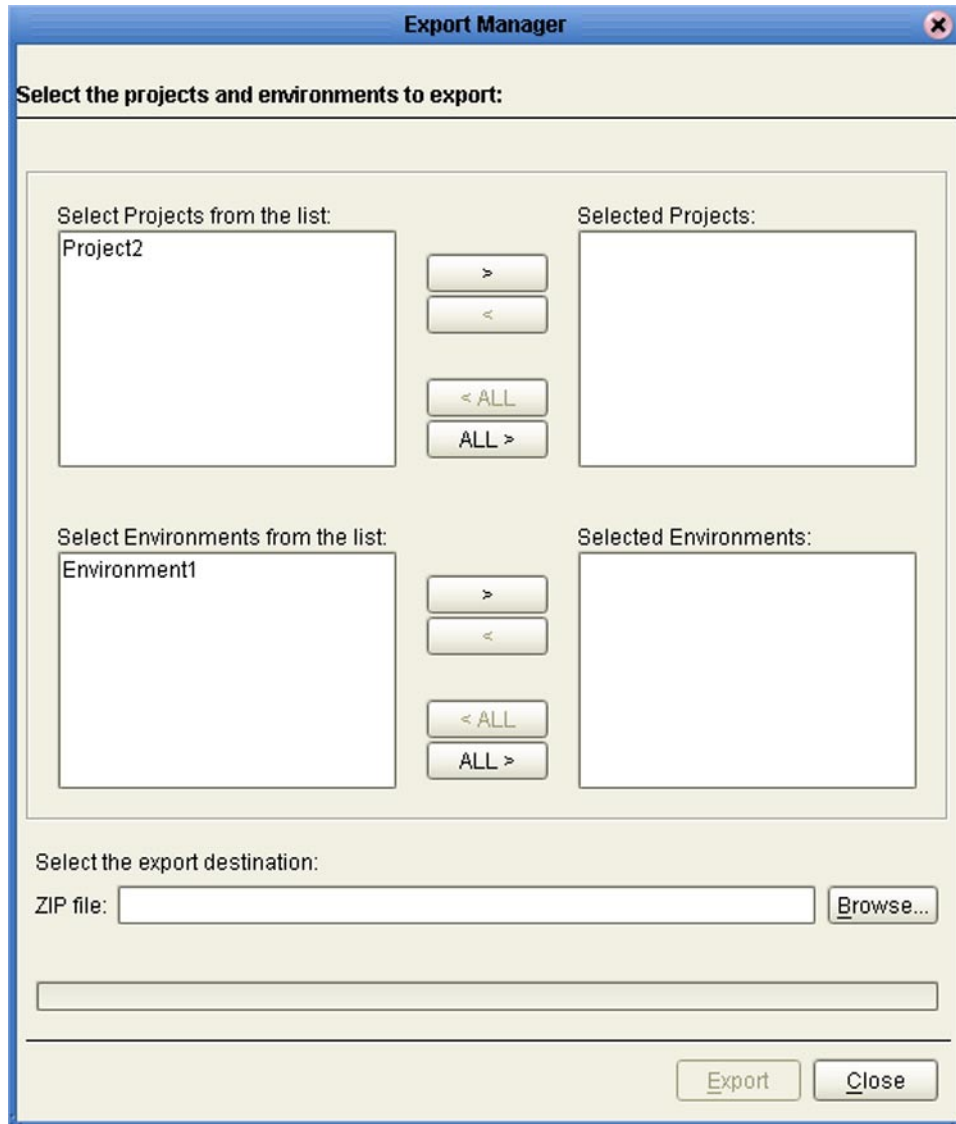
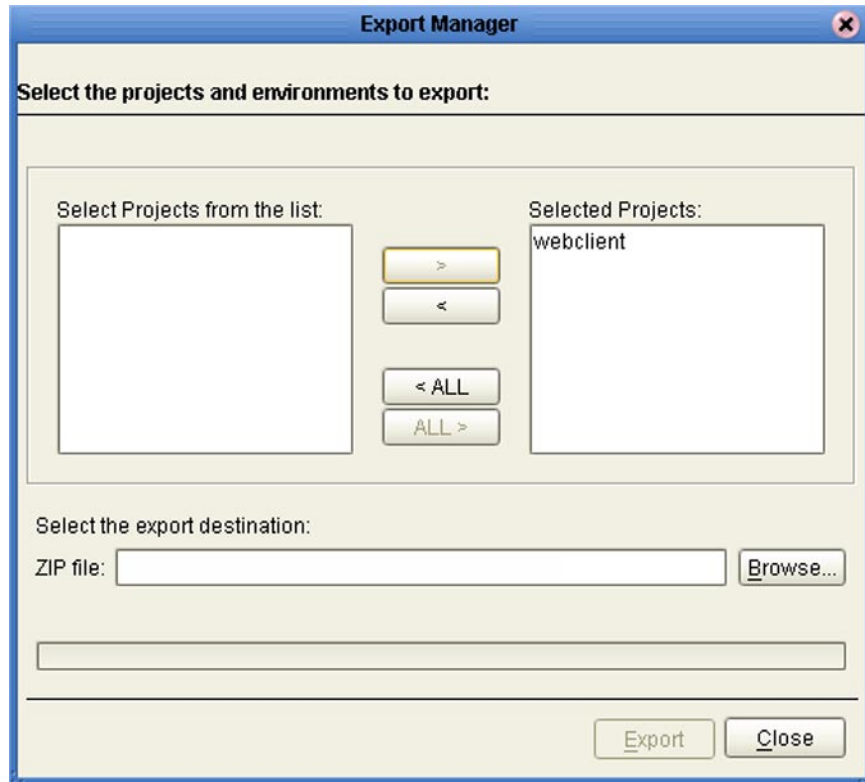


Figure 31 Export Manager Dialog Box (1b)



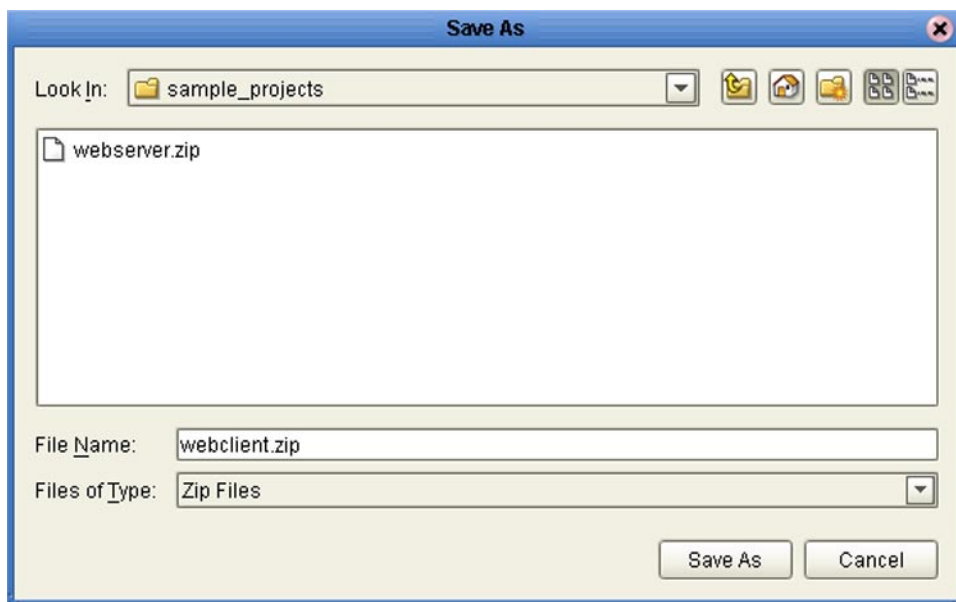
- 2 Highlight the desired Project(s) or Environment(s) in the displayed list, and transfer them to the *Selected Projects* or *Selected Environments* panel using the arrow buttons (see Figure 32).

**Figure 32** Export Manager Dialog Box (2)



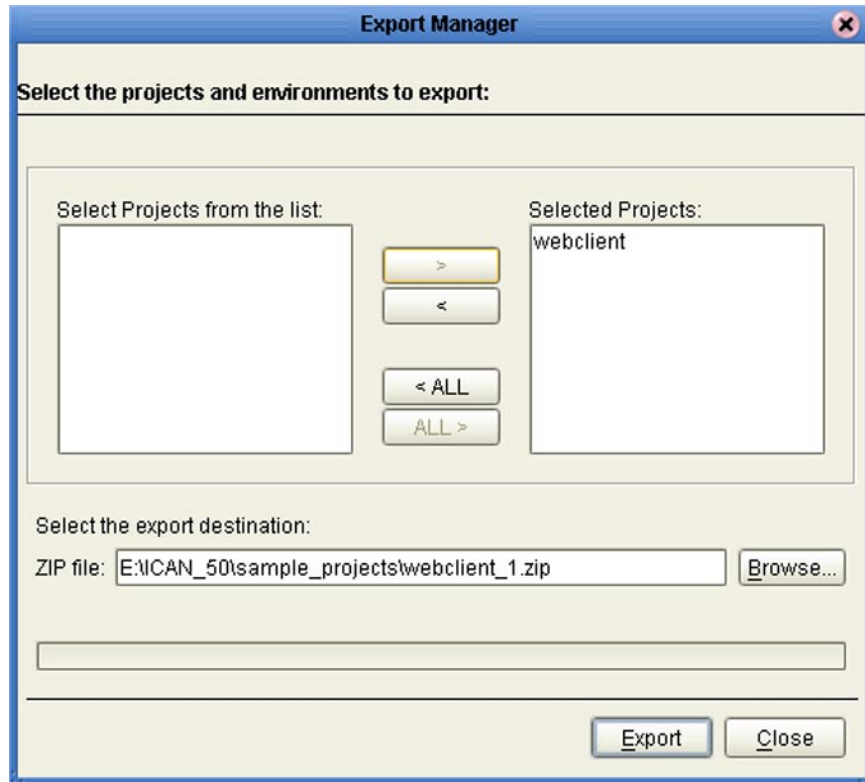
- 3 Click the **Browse** button to display the *Save As* dialog box, as shown in Figure 33.

**Figure 33** Save As Dialog Box



- 4 Select the export destination and change the export file name, if desired.
- 5 Click **Save As** to enter the file name.

**Figure 34** Enter File Name Dialog Box (2)



- 6 Click **Export** to export the Project file (this process may take a few minutes).
- 7 The Export Status message box shown in Figure 35 appears after the file has been exported successfully.

**Figure 35** Export Status Message Box



- 8 Click **OK** to close the message box.

## Exporting a Project Using the Command Line

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

### Location of script file:

*ICAN-root*\repository\util\exportProject.bat (or exportProject.sh)

### Command Syntax:

```
exportProject username password exportfile projectname  
environmentname
```

where:

- ♦ **exportfile** is the name and path for the archive file to contain the Project and/or Environment you are exporting.
- ♦ **projectname** is the name of the Project you are exporting. If you are exporting an Environment only, leave this parameter as an empty string ("").
- ♦ **environmentname** is the name of the Environment you are exporting. If you are exporting a Project only, leave this parameter as an empty string ("")

**Note:** Project names should not include special characters such as a comma (,), single quote ('), or double quote ("). If any of these characters are used, you must escape them when using the exportProject command-line tool.

### To export a Project using the export script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To save the Project **myProject** to the file **c:\project4export.zip**, type:

```
exportProject username password c:\project4export.zip myProject ""
```

### To export an Environment using the export script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To save the Environment **myEnvironment** to the file **c:\environment4export.zip**, type:

```
exportProject username password c:\environment4export.zip ""  
myEnvironment.
```

### To export a Project and an Environment using the export script

- 1 Open a command prompt and change directory to *ICAN-root*\repository\util.
- 2 To save the Project **myProject** and Environment **myEnvironment** to the file **c:\projenv4export.zip**, type:

```
exportProject username password c:\projenv4export.zip myProject  
myEnvironment.
```

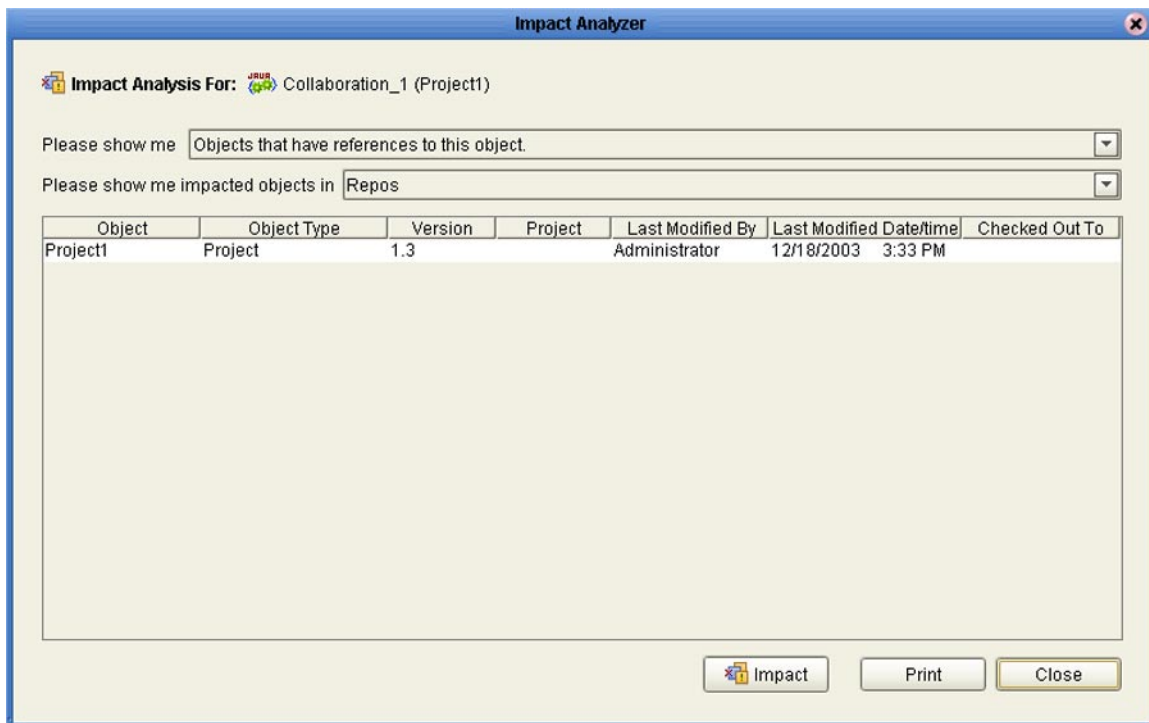
### 4.7.3 Impact Analyzer

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 36.
- 3 In the *Please show me* drop-down list, select items you would like to view. You have the option of viewing either:
  - ♦ Objects that have references to the selected object.
  - ♦ Objects that are referenced by the selected object.
- 4 You can filter the number of listed objects using the *Please show me impacted objects in* drop-down list; the default is the entire **Repository**.
- 5 You can print the object list by clicking **Print** to display the Windows *Print* dialog box.

**Figure 36** Impact Analyzer Dialog Box



**Table 13** Impact Analyzer Command Buttons

Button	Function
 Impact	Performs an impact analysis for the object selected from the object list (not fully implemented).
Print	Displays the Windows <b>Print</b> dialog box, which you can use to print the object list.
Close	Closes the <b>Impact Analyzer</b> dialog box.

## 4.7.4 Version Control

Version control allows you to maintain multiple versions of selected Project or Environment components. The version history of each component is recorded to a log file, and can be viewed by means of a menu option (see [Viewing a Component's Version History](#) on page 65).

**Important:** *More than one person concurrently using the same user ID will circumvent this version control system, and one person's work can be overwritten by another. You should ensure that all personnel using Enterprise Designer use unique IDs.*

### Checked-In State

When a component is checked in to the version control system, it is locked against modification until being checked out, and a lock is displayed in the component's icon in the Enterprise Explorer. Figure 37 shows the lock, using the OTD icon as an example. See [Checking a Component In](#) on page 66, [Checking a Component In Without Revisions](#) on page 67, and [Checking In a Previous Version as the Latest Version](#) on page 71.

**Figure 37** *Checked In Icon (OTD Example)*



### Checked-Out State

When the latest version of a component is checked out from the version control system, it is locked against another user checking it out. A writing pad icon (see Figure 38) is displayed next to the component's icon in the Enterprise Explorer, indicating that it is checked out. See [Checking a Component Out](#) on page 68.

**Figure 38** *Checked Out Icon*



### Retrieved State

When any version of a component is retrieved from the version history dialog box, it is *not* locked against another user checking it out or retrieving it. A combined writing pad/warning icon (see Figure 39) is displayed next to the component's icon in the Enterprise Explorer, indicating that it is in your workspace—but warning you that it is not locked in any way. See [Retrieving a Component to Your Workspace](#) on page 69 and [Checking In a Previous Version as the Latest Version](#) on page 71.

**Figure 39** *Retrieved Icon*



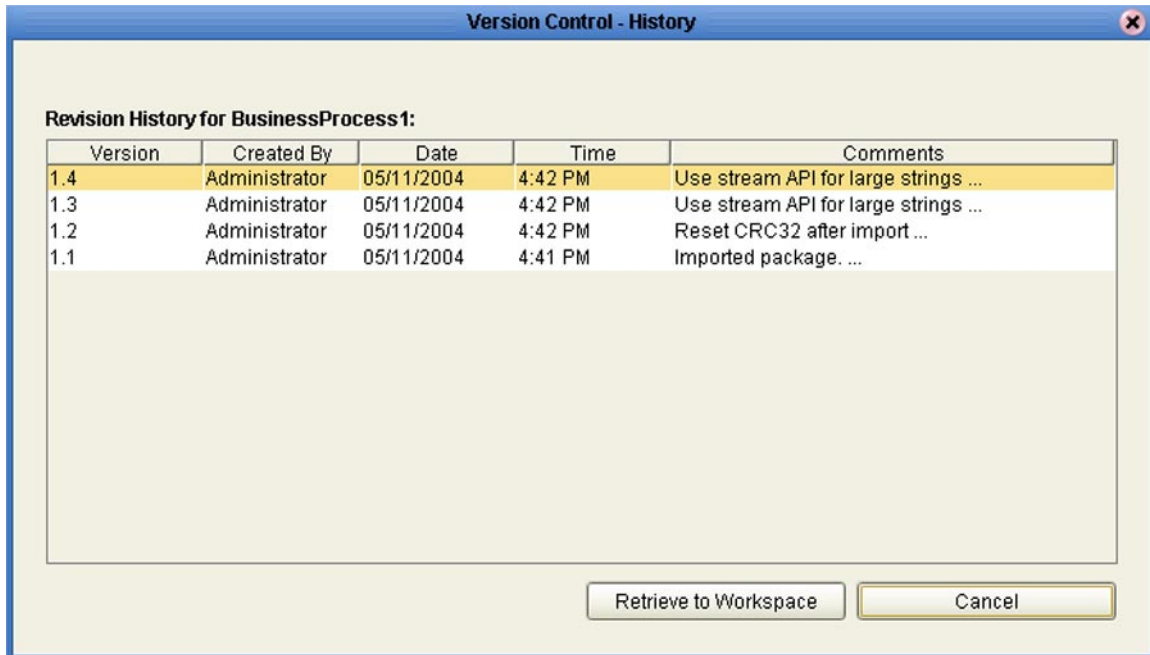


## Viewing a Component's Version History

To view the version history for a component

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Select **Version History** to display the *Version Control - History* information box shown in Figure 40.

**Figure 40** Version Control - History Dialog Box



- 3 Right-click in the *Comments* column to display the full text of the comment.
- 4 Click **Cancel** to close the box.

**Note:** *If a version is checked out or retrieved to a user's workspace, the appropriate icon also appears in the Version column.*

**Note:** *The version history for a component that has been **cut** and pasted is preserved, since there can be only one instance of it. The version history for a component that has been **copied** and pasted is **not** preserved, since there can be many instances of it; the version number for each pasted instance is reset.*

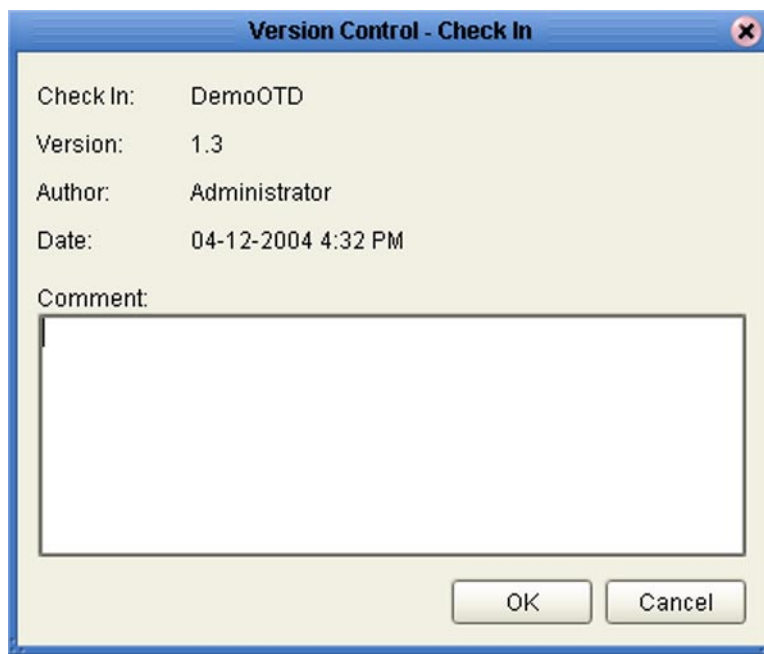
## Checking a Component In

Once you have created and configured a component for the first time, or created a revised version of an existing component, you must check that component in to save it to the common area of the Repository and release your lock on the object.

### To check in a new version of a Project or Environment component

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Select **Check In** to display the *Version Control - Check In* dialog box shown in Figure 41.

**Figure 41** Version Control - Check In Dialog Box



- 3 Type in a description of the changes in the new version.
- 4 Click **OK** to check the new version in.

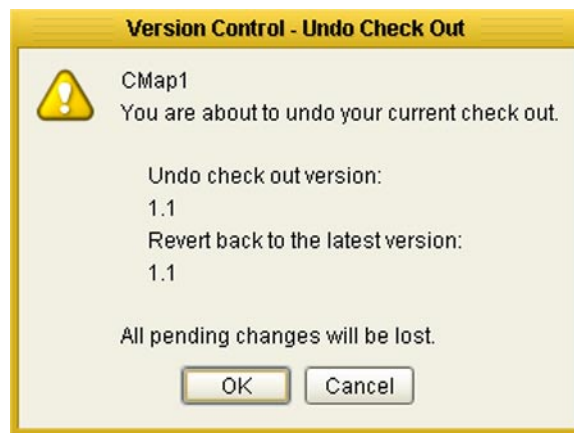
## Checking a Component In Without Revisions

When you have checked the latest version of a component out and want to check it back in without any revisions, you can simply cancel the check-out by using the following procedure. The version number is not incremented.

### To check in a Project or Environment component without revisions

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Select **Undo Check Out** to display the *Version Control - Undo Check Out* dialog box shown in Figure 42. (This option is not available for all components.)

**Figure 42** Version Control - Undo Check Out Dialog Box



- 3 Click **OK** to check the currently checked-out version back in.

**Note:** *This procedure is also valid for retrieved versions.*

## Checking a Component Out

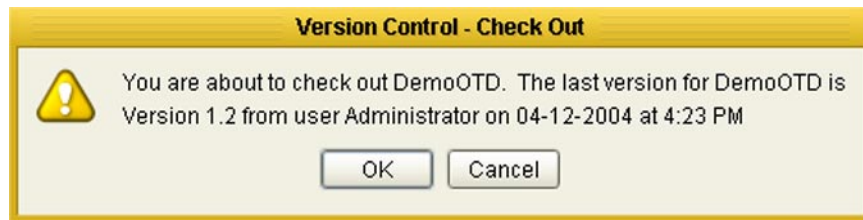
You can check out the current version of a component by using the following procedure.

**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.*

### To check out the latest version of a Project or Environment component

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Select **Check Out** to display the *Version Control - Check Out* dialog box shown in Figure 43.

**Figure 43** Version Control - Check Out Dialog Box



- 3 Click **OK** to check the component out.
- 4 Click the **Save** or **Save All** icon to place the version in your Repository workspace.

## Retrieving a Component to Your Workspace

You can retrieve either the current or a previous version of any of the following components by retrieving it from the Version History information box.

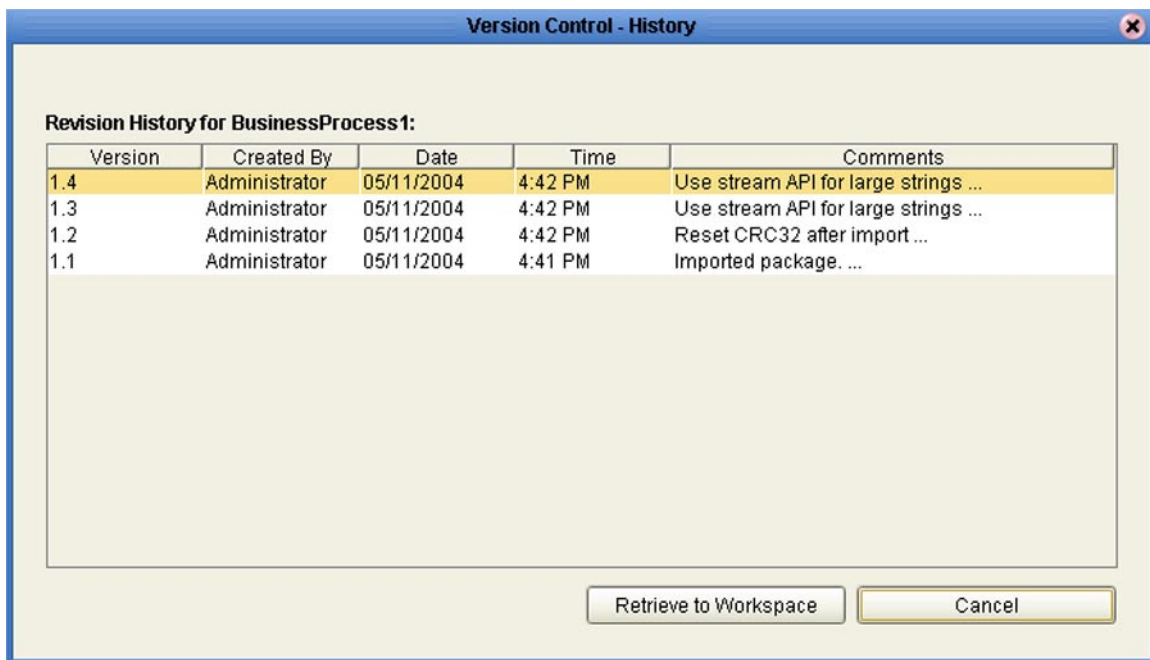
- Object Type Definition (DTD)
- Object Type Definition (XSD)
- eInsight Business Process

Retrieving does *not* lock the file from being checked out or retrieved by other users for editing. To check a retrieved version back in as the latest version, you must use the **Make Latest** option described in [Checking In a Previous Version as the Latest Version](#) on page 71.

### To retrieve an older version of a Project or Environment component

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Select **Version History** to display the *Version Control - History* dialog box shown in Figure 40.

**Figure 44** Version Control - History Dialog Box



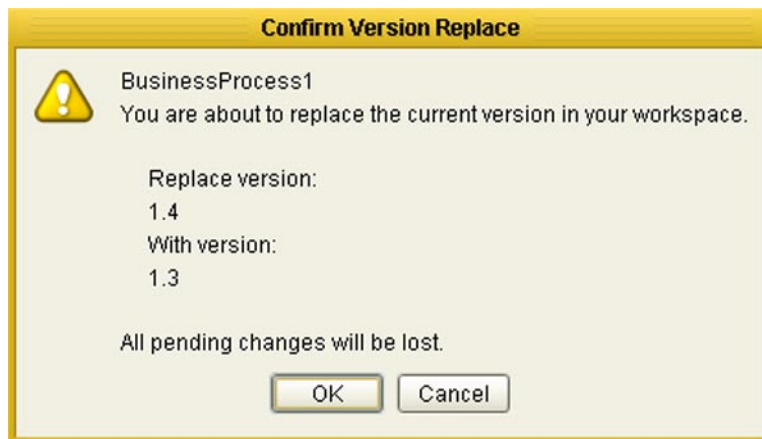
- 3 Select the version you want to retrieve and click **Retrieve to Workspace**.
  - A If you are attempting to retrieve the *latest* version of the component, you will be presented with the dialog box shown in Figure 45.

**Figure 45** Access File Dialog Box



- ♦ **Check Out for Edit** copies the file to your workspace and locks it—the file becomes read-only to other users. This is the same mechanism as described in [Checking a Component Out](#) on page 68. To check the latest version out from the dialog box, you must select this option and click **OK**. You will then be presented with the dialog box shown previously (Figure 43).
  - ♦ **Retrieve to Workspace** copies the file to your workspace, but does not prevent it from being checked out or retrieved by other users. This is the default setting for the dialog box; simply click **OK**.
- B** If you are attempting to retrieve a *previous* version of the component, you will be presented with the dialog box shown in Figure 46. Clicking **OK** will overwrite any other version you have retrieved to your workspace, or replace the currently checked-in version in your workspace only—other users will be unaffected.

**Figure 46** Confirm Version Replace Dialog Box



**Note:** *If you have the latest version of the component checked out to your workspace, the **Checked Out** icon will appear in the Version column of the Version History dialog and the **Retrieve to Workspace** button will be disabled. You must check the latest version back in to version control before you can retrieve any version.*

## Checking In a Previous Version as the Latest Version

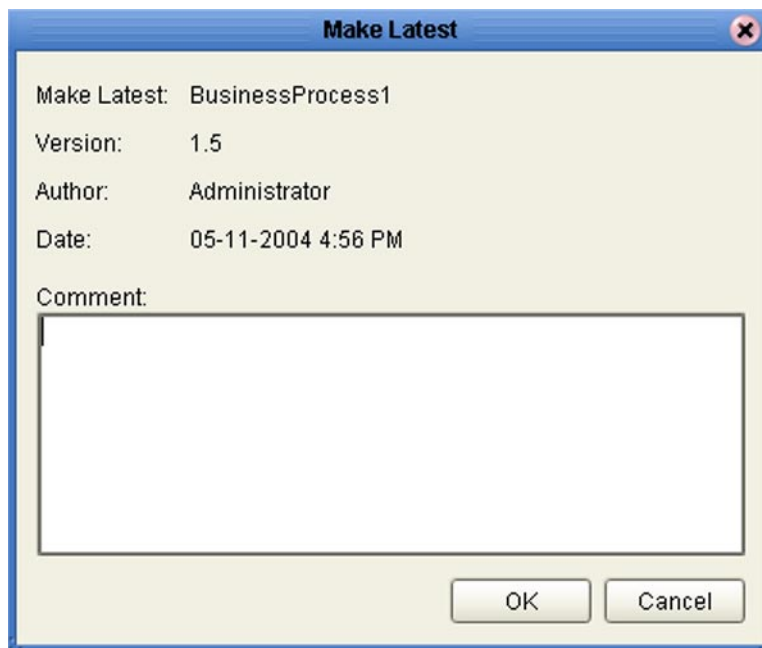
If you have retrieved a previous version of any of the following components to your workspace, you can check it in to the version control system as the latest version by selecting the *Make Latest* option.

- Object Type Definition (DTD)
- Object Type Definition (XSD)
- eInsight Business Process

### To make a previous version of a component become the latest version

- 1 In the Enterprise Explorer, select the component and right-click to display its context menu.
- 2 Click **Make Latest** to display a confirmation dialog box.
  - A If the latest checked-in version of the component has not changed since you retrieved the previous version, you will see the dialog box shown in Figure 46.

**Figure 47** Make Latest Dialog Box



Type in a description of the changes in this version and click **OK**. The version in your workspace will be checked in as the latest version of the component.

- B If the latest checked-in version of the component is different from the one that was current when you retrieved the previous version, you will first see the dialog box shown in Figure 48.

**Figure 48** Confirm Latest Version Override Dialog Box



If you are sure you want to replace the current latest version, click **OK** to display the dialog box shown previously (Figure 47). Type in a description of the changes in this version and click **OK**. The version in your workspace will be checked in as the latest version of the component.

**Important:** *This situation can occur if another user has made changes to the latest version—you must use caution when checking in your version, since the other user's changes will be superseded.*

## Command-line Utilities

If you encounter problems with the version control system, there are two command-line utilities—a Repository version control utility and a workspace cleanup script—that can be run by personnel with Administrator privileges. These utilities should be used as a last resort, and with the utmost caution. See the *eGate Integrator System Administration Guide* for information.



# ICAN Projects

This chapter describes the components of an ICAN Project incorporating an eInsight Business Process, and the use of the Enterprise Designer in defining your Project.

## What's in This Chapter

- [Overview](#) on page 73
- [Project Explorer](#) on page 74
- [Creating a Project](#) on page 84
- [Using the Connectivity Map Editor](#) on page 86
- [External Applications](#) on page 88
- [Component Connections](#) on page 91
- [Defining Constants and Variables](#) on page 94

---

## 5.1 Overview

An eInsight 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 123). Components developed for use in one Project can be used in another, and a Project can internally reference another Project.

**Important:** *Because of a Java limitation, Project activation can fail on Windows if the path is too long. SeeBeyond recommends that you keep your Projects paths relatively short.*

### 5.1.1 Project Components

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

- [Using the Connectivity Map Editor](#) on page 86
- [External Applications](#) on page 88
- [Component Connections](#) on page 91

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

- **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 eInsight Enterprise Service Bus, and the methods available for operating on them, and also interactions with external APIs. See [Object Type Definitions](#) on page 97.

---

## 5.2 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 49).

**Figure 49** Project Explorer














The Project Explorer is used in conjunction with the various editors to create and configure the components of a Project. Each component in the Project Explorer has an icon to identify the component type (see [Project Explorer Icons](#) on page 75). Right-clicking on a component displays a context menu for that component (see [Context Menus](#) on page 76), from which you can select appropriate actions.

**Note:** *Select **Refresh All from Repository** before you open any Project component (such as a service) to ensure that you open the latest version of the 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 <b>Repository</b> , which is the central ICAN database where all Project information is saved. Binary files required at run time are also stored here.
	Represents the <b>Project</b> or subproject.
	Represents the SeeBeyond <b>Default Project</b> or subproject.
	Represents a <b>Business Process</b> . A lock displayed in the lower-left corner indicates that the Business Process is currently checked into the version control system (see OTD example).
	Represents a <b>Connectivity Map</b> , which contains the business logic and information about the data transmission. A lock displayed in the lower-left corner indicates that the Connectivity Map is currently checked into the version control system (see OTD example).
	Represents a <b>Project variable</b> or <b>constant</b> . A lock displayed in the lower-left corner indicates that the variable or constant is currently checked into the version control system (see OTD example).
	Represents an <b>Object Type Definition (OTD)</b> file.
	A lock displayed in the lower-left corner indicates that the OTD is currently checked into the version control system.
	Represents a <b>Deployment Profile</b> , which specifies how Project components will be deployed in an Environment. A lock displayed in the lower-left corner indicates that the Deployment Profile is currently checked into the version control system (see OTD example).
	Displayed along side one of the above icons, indicates that the current latest version of the component has been checked out for editing.
	Displayed along side one of the above icons, indicates that some version of the component—either the latest version or a previous version—has been retrieved to the local workspace.

## 5.2.2 Context Menus

Right-clicking a component in the Project Explorer displays a context menu for that component. Only those menu options that are allowed for the component in its current state are activated.

### Repository Menu

**Figure 50** Repository Menu



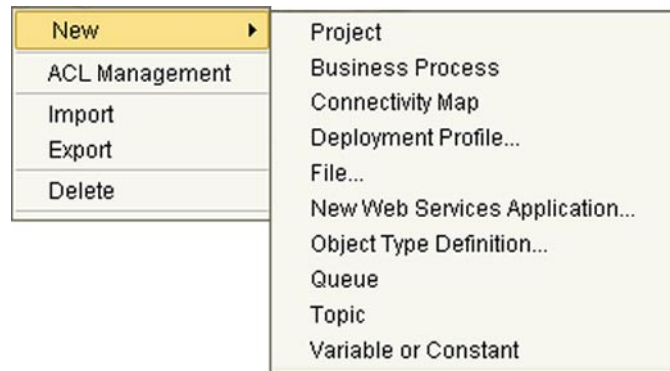
**Table 15** Repository Menu Options

Option	Function
Project	Adds a new Project to the Repository.
Sort by Type	Places all objects in order by grouping object types.
Sort by Name	Places all objects in alphabetical order.
Sort by Date	Places all objects in order by creation date, from oldest to newest.
Import	Displays a dialog box with which you can import a Project or Environment into the Repository.
Export	Displays a dialog box with which you can export a Project or Environment from the Repository to an archive file.
Refresh All from Repository	Refreshes the Project Explorer to display the current contents of the Repository. (Open editors are not refreshed.)
User Management	Displays the User Management dialog box, where an Administrator can manage user access to the Repository with options for adding, modifying, and deleting users. See the <i>eGate Integrator System Administration Guide</i> .
Properties	Displays a dialog box showing the configuration properties of your Repository. See the <i>eGate Integrator System Administration Guide</i> .

**Note:** Project names should not include special characters such as a comma (,), single quote ('), or double quote ("). If any of these characters are used, you must escape them when using the `import/exportProject` command-line tool.

## Project Menu

**Figure 51** Project Menu



**Table 16** Project Menu Options

Option	Option	Function
New	Project	Adds a Subproject folder to the selected Project.
	Business Process	Displays the user interface for creating a new Business Process.
	Connectivity Map	Adds a Connectivity Map to the Project. See <a href="#">Using the Connectivity Map Editor</a> on page 86.
	Deployment Profile	Displays a dialog box with which you can create a Deployment Profile for the selected Project. See <a href="#">Deployment Editor</a> on page 175.
	File	If the File eWay is installed, displays a dialog box with which you can create an external file to use with the Project. (This is an example of an External Application—the applications that are displayed in the menu depend upon which eWays are installed on your system.)
	New Web Services Appl.	Adds a Web Services application to the selected Project. See <a href="#">Web Services Application</a> on page 196.
	Object Type Definition	Displays the <b>OTD Wizard</b> , with which you can create an Object Type Definition (OTD) file. See <a href="#">Using the OTD Wizards</a> on page 98 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 an Administrator can assign read/write/delete privileges to users for the selected Project. See the <i>eGate Integrator System Administration Guide</i> .

**Table 16** Project Menu Options

Option	Option	Function
Import		Displays a dialog box with which you can import a Project as a Subproject under the selected Project. See <a href="#">Project/Environment Import</a> on page 53.
Export		Displays a dialog box with which you can export the selected Project. See <a href="#">Project/Environment Export</a> on page 57.
Rename		Activates the field, allowing you to rename the selected Project.
Delete		<p>Deletes the selected Project, subject to the following conditions:</p> <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the Project (see <i>ACL Management</i>, above).</li> <li>▪ The Project is not checked out by anyone other than yourself.</li> </ul> <p>If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected Project. Clicking <b>Yes</b> then deletes the Project.</p> <p>If the selected Project contains active Deployment Profiles, a dialog box is displayed listing those deployments and requesting that you deactivate them. You must select deactivation to enable the <b>OK</b> button.</p>

**Note:** Subproject names should not include special characters such as a comma (,), single quote ('), or double quote ("). If any of these characters are used, you must escape them when using the `import/exportProject` command-line tool.

## Connectivity Map Menu

**Figure 52** Connectivity Map Menu

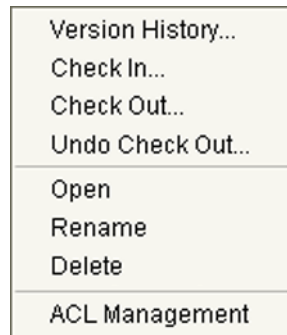
Open
ACL Management
Version History...
Check In...
Check Out...
Undo Check Out...
Rename
Delete

**Table 17** Connectivity Map Menu Options

Command	Function
Open	Opens the Connectivity Map Editor for the selected Connectivity Map. See <a href="#">Using the Connectivity Map Editor</a> on page 86.
ACL Management	Displays the ACL Properties dialog box, with which an Administrator can assign read/write/delete privileges to users for the selected Connectivity Map. See the <i>eGate Integrator System Administration Guide</i> .
Version History	Displays a dialog box with which you can track the version history for the selected Connectivity Map. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Check In	Displays a dialog box with which you can check in a new version of the selected Connectivity Map. See <a href="#">Checking a Component In</a> on page 66 for more details.
Check Out	Displays a dialog box with which you can check out the current version of the selected Connectivity Map. See <a href="#">Checking a Component Out</a> on page 68 for more information.
Undo Check Out	Displays a dialog box with which you can undo the check-out of the selected Connectivity Map. See <a href="#">Checking a Component In Without Revisions</a> on page 67 for more information.
Rename	Activates the field, allowing you to rename the selected Connectivity Map.
Delete	Deletes the selected Connectivity Map, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the Connectivity Map (see <i>ACL Management</i>, above).</li> <li>▪ The Connectivity Map is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected Connectivity Map. Clicking <b>Yes</b> then deletes the Connectivity Map.

## Deployment Profile Menu

**Figure 53** Deployment Profile Menu



**Table 18** Deployment Profile Menu Options

Command	Function
Version History	Displays a dialog box with which you can track the version history for the selected Deployment Profile. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Check In	Displays a dialog box, with which you can check in a new version of the selected Deployment Profile. See <a href="#">Checking a Component In</a> on page 66 for more details.
Check Out	Displays a dialog box with which you can check out the current version of the selected Deployment Profile. See <a href="#">Checking a Component Out</a> on page 68 for more information.
Undo Check Out	Displays a dialog box with which you can undo the check-out of the selected Deployment Profile. See <a href="#">Checking a Component In Without Revisions</a> on page 67 for more information.
Open	Opens the Deployment Editor, showing the selected Deployment Profile. See <a href="#">Deployment Editor</a> on page 175.
Rename	Activates the field, allowing you to rename the selected Deployment Profile.
Delete	Deletes the selected Deployment Profile, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the Deployment Profile (see <i>ACL Management</i>, above).</li> <li>▪ The Deployment Profile is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected Deployment Profile. Clicking <b>Yes</b> then deletes the Deployment Profile.
ACL Management	Displays the ACL Properties dialog box, with which an Administrator can assign read/write/delete privileges to users for the selected Deployment Profile. See the <i>eGate Integrator System Administration Guide</i> .



## File Menu

**Figure 54** File Menu



**Table 19** File Menu Options

Command	Function
Version History	Displays a dialog box with which you can track the version history for the selected file. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Check In	Displays a dialog box, with which you can check in a new version of the selected file. See <a href="#">Checking a Component In</a> on page 66 for more details.
Check Out	Displays a dialog box with which you can check out the current version of the selected file. See <a href="#">Checking a Component Out</a> on page 68 for more information.
Update	Displays a file browser dialog box with which you can replace the file with an updated version.
Export	Displays a file browser dialog box with which you can save the selected file to another directory.
Delete	Deletes the selected file, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the file (see <i>ACL Management</i>, above).</li> <li>▪ The file is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected file. Clicking <b>Yes</b> then deletes the file.

## Object Type Definition Menu

**Figure 55** OTD Menu



**Table 20** OTD Menu Options

Command	Function
Open	Opens the OTD Editor, showing the selected Object Type Definition. See <a href="#">OTD Editor</a> on page 100.
Relaunch	Relaunches the DTD or XSD OTD Wizard, so that you can re-define the selected Object Type Definition while retaining the original OID. The following conditions apply: <ul style="list-style-type: none"> <li>▪ You must have <i>write</i> privileges for the OTD (see <i>ACL Management</i>, above).</li> <li>▪ You must select the same top level node when relaunching the OTD.</li> <li>▪ The OTD must not be checked out by anyone other than yourself.</li> <li>▪ The OTD must not be imported from another Project.</li> <li>▪ The OTD must have been created in eGate Integrator 5.0.4 (or later release).</li> </ul>
Version History	Displays a dialog box with which you can track the version history for the selected Object Type Definition. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Check In	Displays a dialog box, with which you can check in a new version of the selected Object Type Definition. See <a href="#">Checking a Component In</a> on page 66 for more details.
Check Out	Displays a dialog box with which you can check out the current version of the selected Object Type Definition. See <a href="#">Checking a Component Out</a> on page 68 for more information.
Undo Check Out	Displays a dialog box with which you can undo the check-out of the selected Object Type Definition. See <a href="#">Checking a Component In Without Revisions</a> on page 67 for more information. <b>Note:</b> Does not apply to WSDL OTDs.
Make Latest	Allows you to check in the version of the selected Object Type Definition that was retrieved to your workspace, making it the latest version. See <a href="#">Checking In a Previous Version as the Latest Version</a> on page 71. <b>Note:</b> Does not apply to WSDL OTDs.

**Table 20** OTD Menu Options

Command	Function
ACL Management	Displays the ACL Properties dialog box, with which an Administrator can assign read/write/delete privileges to users for the selected Object Type Definition. See the <i>eGate Integrator System Administration Guide</i> .
Delete	Deletes the selected Object Type Definition, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the OTD (see <i>ACL Management</i>, above).</li> <li>▪ The OTD is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected OTD. Clicking <b>Yes</b> then deletes the OTD.
Rename	Activates the field, allowing you to rename the selected Object Type Definition.

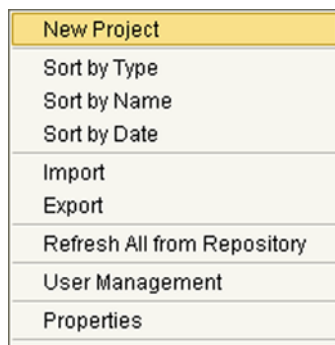
## 5.3 Creating a Project

You can create an ICAN Project by using the context menu system in Enterprise Explorer together with the Connectivity Map Editor. This section contains a brief outline of the procedure to get you started—an end-to-end description can be found in the *eInsight Enterprise Service Bus User's Guide*.

To create an ICAN Project

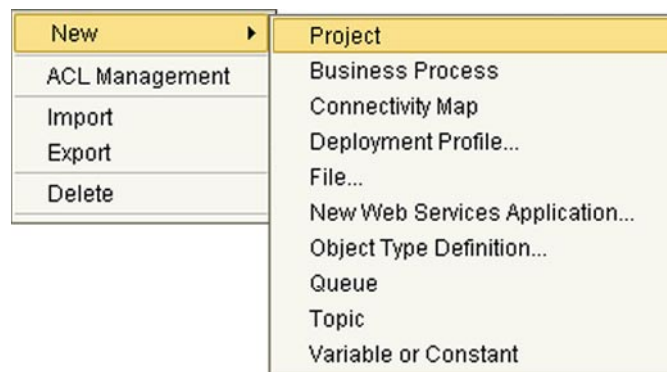
- 1 Right-click the **Repository** in the Project Explorer to display its context menu (see Figure 56).

**Figure 56** Repository Context Menu - New Project



- 2 Select a **New Project**, which initiates a hierarchical structure under the Repository.
- 3 Rename the Project as desired.
- 4 Right-click the **Project** to display its context menu (see Figure 57).

**Figure 57** Project Context Menu - New Project



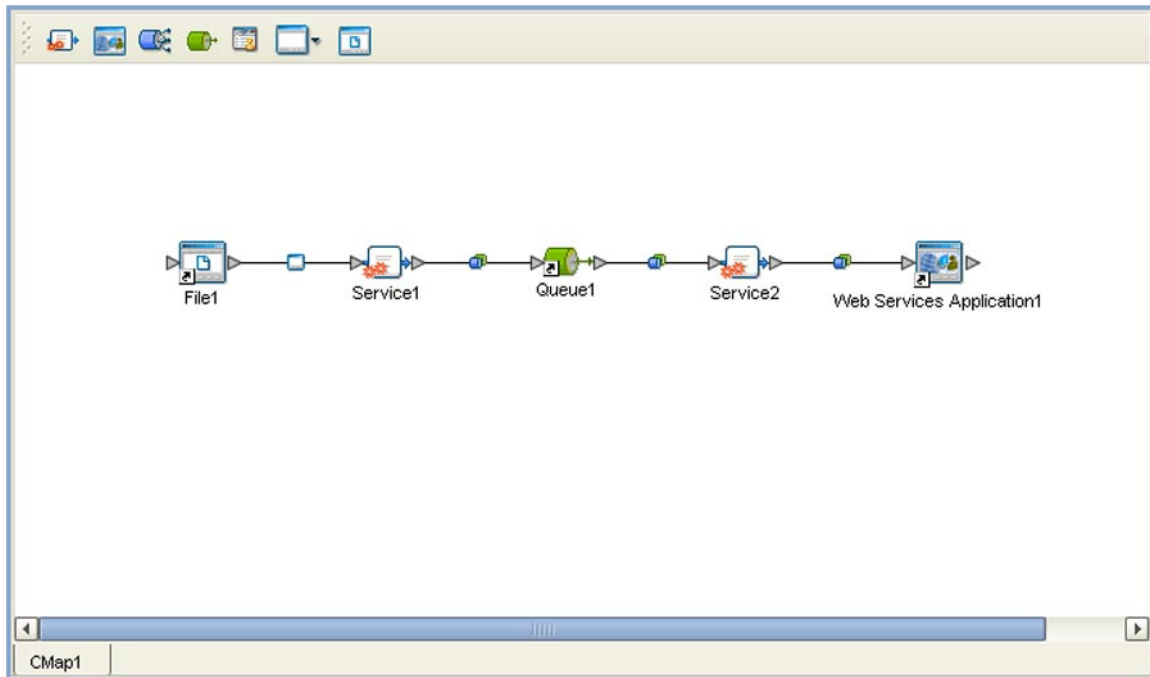
- 5 Select a **New Connectivity Map**, which displays the Connectivity Map Editor.
- 6 In the Connectivity Map Editor, populate the canvas with the various components needed for your Project (see [Using the Connectivity Map Editor](#) on page 86), but leave the connectivity for later.
- 7 After you have developed the basic architecture of your Project, return to the Project context menu in Project Explorer.

- 8 Select a **New Business Process**.
- 9 Select a **New Object Type Definition**, and build your OTD as described in **Object Type Definitions** on page 97.
- 10 Return to the Connectivity Map and connect the various components (see **Using the Connectivity Map Editor** on page 86).
- 11 Save and test your Project.

## 5.4 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 58). 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 58** Connectivity Map Window



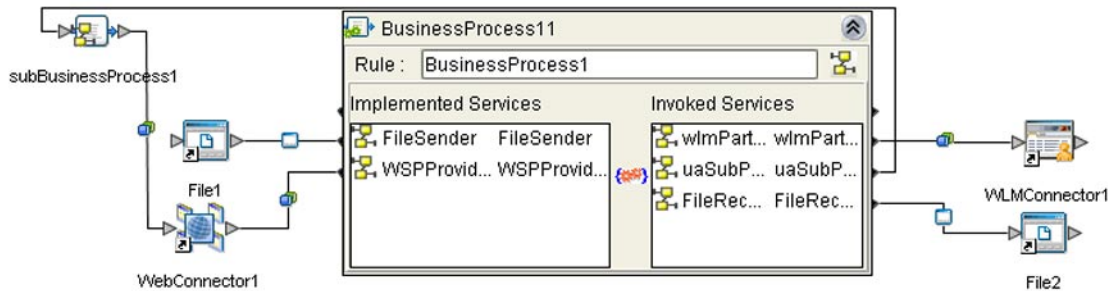
The drag-and-drop components include services, queues, topics, schedulers, and external applications. Additional components, such as eWays and JMS Clients, are placed automatically when you link the components you have placed manually.

A service provides a framework for a business process, which contains the information required to execute a set of business rules. See the *eInsight Enterprise Service Bus User's Guide* for information regarding business processes.

A *message destination* is a container for stored data, and can follow either the topic or queue JMS model. A *topic* is a message destination that conforms to the publish-and-subscribe messaging paradigm. A *queue* is a message destination that conforms to the point-to-point messaging paradigm. See the *eGate Integrator JMS Reference Guide* for detailed information on message destinations.







When there are multiple destinations, the Connectivity Map Editor cannot resolve which output port connects to which destination. Because of this, the Business Process must be defined first, and the connections must be drawn by opening the Business Process Binding box in Connectivity Map (see Figure 59).

**Figure 59** Linking Multiple Message Destinations



The Connectivity Map Editor toolbar contains the icons listed in Table 21, plus additional icons representing eInsight Enterprise Service Bus add-ons and other ICAN components that you may have installed.

**Table 21** Connectivity Map Toolbar Icons

Icon	Component	Function
	Service	A logical component that provides the framework for a business process. See the <i>eInsight Enterprise Service Bus User's Guide</i> for information.
	Queue	A Message Destination that conforms to the point-to-point messaging paradigm, having one sender and one receiver. See the <i>eGate Integrator JMS Reference Guide</i> for information.
	Topic	A Message Destination that conforms to the publish/subscribe messaging paradigm, having one sender (publisher) and multiple receivers (subscribers). See the <i>eGate Integrator JMS Reference Guide</i> for information.
	Web Services Application	Represents a Web Services application (see <a href="#">Web Services Application</a> on page 196).
	External Applications	Represents an application external to eInsight Enterprise Service Bus. Click the arrow beside the icon to view a list of specific applications to which you can connect. See <a href="#">External Application Drop-Down Menu</a> on page 88.
	Scheduler	Represents a scheduling component of the Connectivity Map. Use this component to set data transfer to occur at set intervals. See <a href="#">Schedulers</a> on page 89.

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.

---

## 5.5 External Applications

The basic purpose of eInsight Enterprise Service Bus 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 eInsight Enterprise Service Bus 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 60.

**Figure 60** External Application Drop-Down Menu



Selecting the check box beside an individual external application adds that icon to the toolbar; clearing the check box removes it from the toolbar.



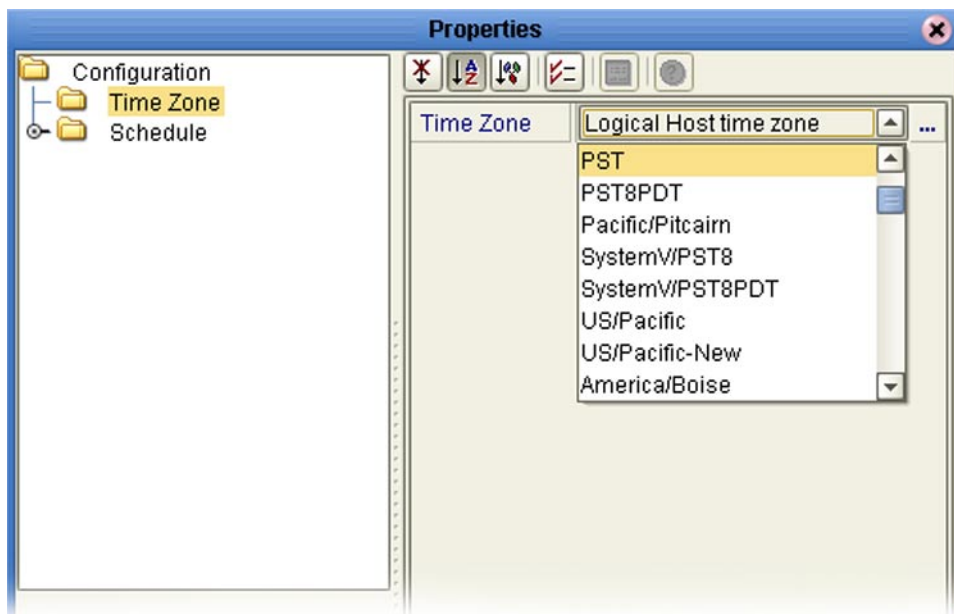
## 5.5.1 Schedulers

A Scheduler allows a service to be performed at a prescribed interval. Once the scheduler is connected to a service in the Connectivity Map, double-clicking the JMS Client (the connector icon) displays the *Properties* dialog box for that scheduler. Double-clicking the properties field enables it for editing.

### Time Zone

Selecting **Time Zone** displays the *Time Zone* property field in which you specify your local time zone, so that your schedule will be synchronized to the local time (see Figure 61).

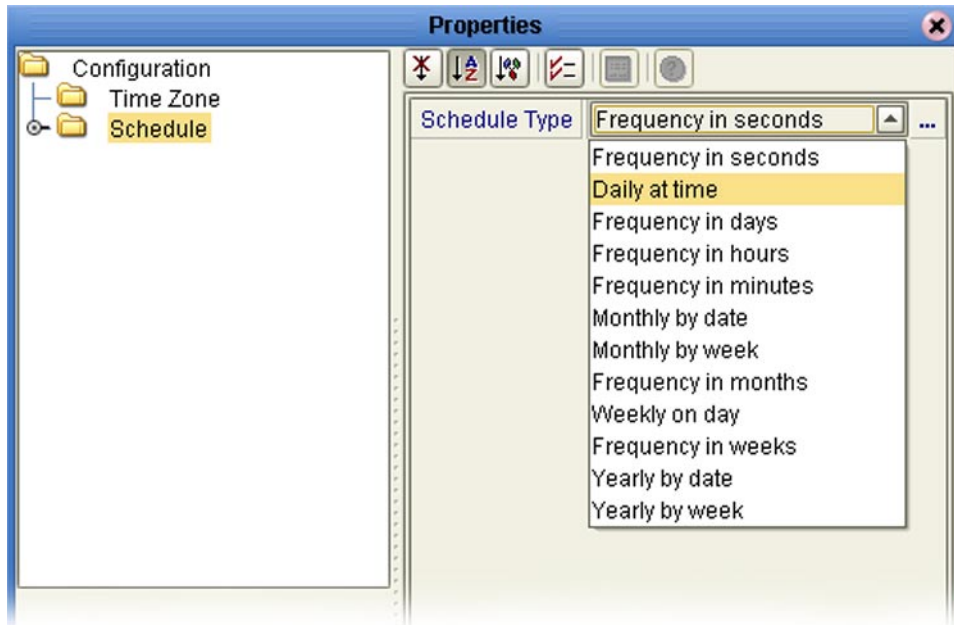
**Figure 61** Scheduler Properties Dialog Box - Time Zone



### Schedule Type

Selecting **Schedule** displays the *Schedule Type* property field which you set to the type of schedule you want to use (see Figure 62).

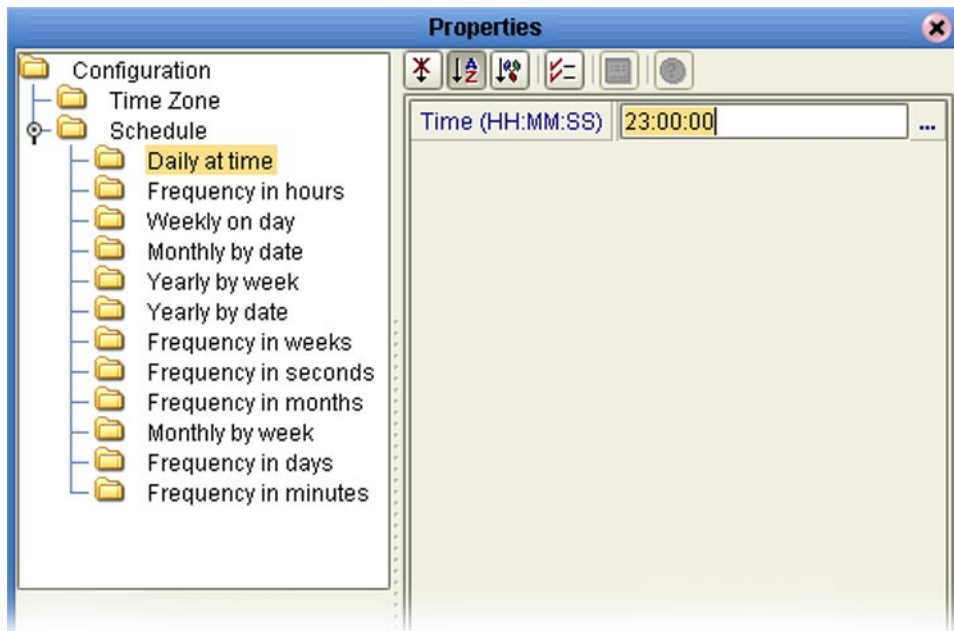
**Figure 62** Scheduler Properties Dialog Box - Schedule Type



## Schedule Value

Selecting a **Schedule Type** node in the explorer tree displays the property field for that schedule type, in which you specify the desired value (see Figure 63).

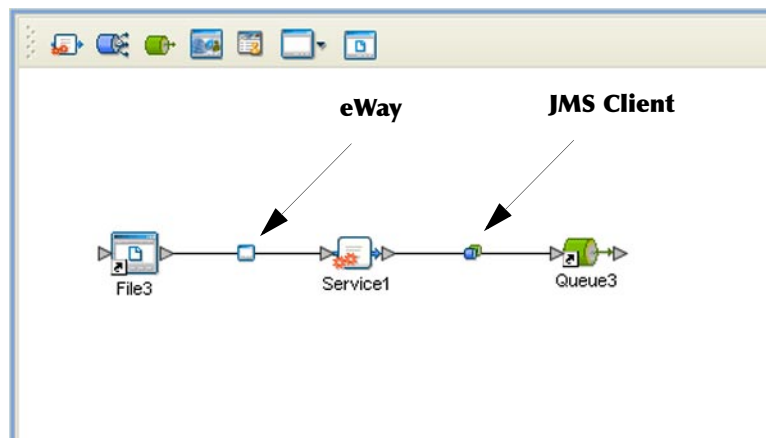
**Figure 63** Scheduler Properties Dialog Box - Schedule Value



## 5.6 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 64).

**Figure 64** Connection Icons in a Connectivity Map

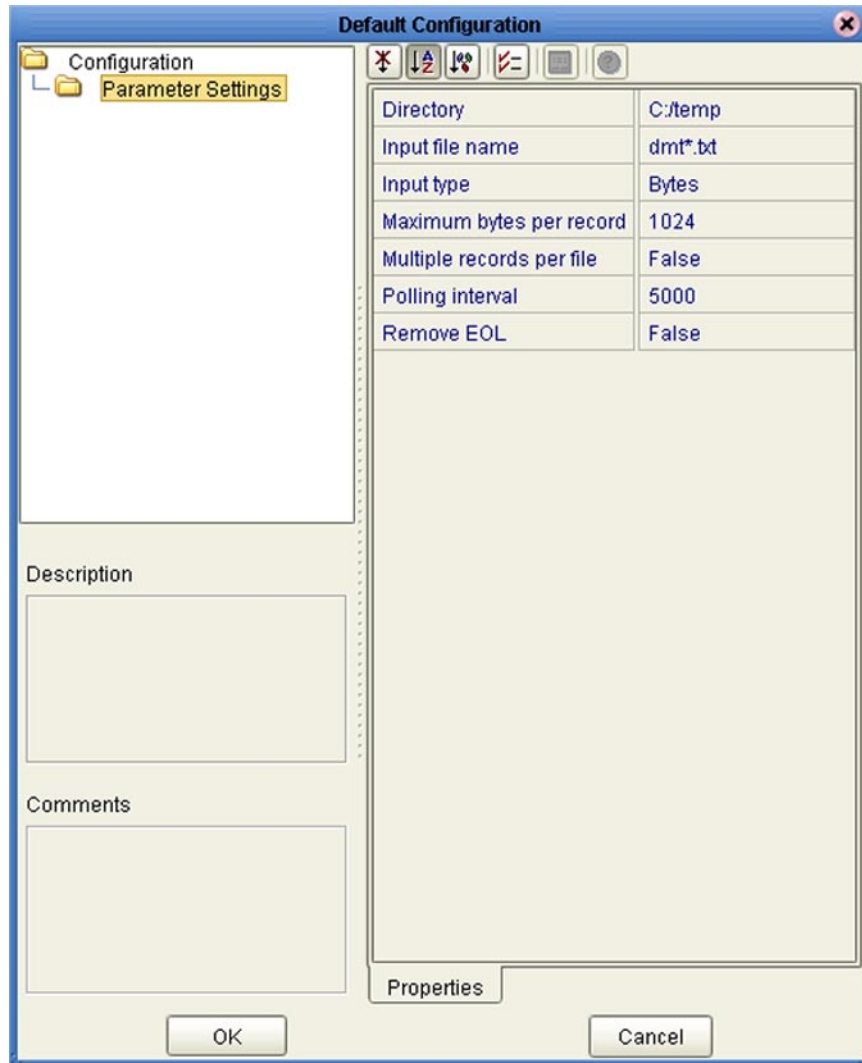


- When you link an external application with a Business Process, the Enterprise Designer automatically adds an eWay Connection icon to the link. The eWay enables communication and movement of data between the external application and the eInsight Enterprise Service Bus system. The eWay configuration 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.

### 5.6.1 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 65 shows a dialog box that lists the configuration properties for a File eWay.

**Figure 65** 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. 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 22.
- The **Description** box contains a brief description of the contents of the item currently selected in the Configuration Tree.
- The **Comments** box is for user comments about the item selected in the Configuration Tree.

**Table 22** 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 <b>Customizer</b> 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 Defining Constants and Variables

You can define both constants and variables for use in a specific Project.

### Project Constants

Constants are name/value pairs that are visible across the Project. For example, Figure 66 shows a standard currency defined to be used globally throughout the system.

**Figure 66** Creating a Project Constant

**Table 23** Project Variable Options

Property	Option	Description/Usage
Name		Your name for the Project constant.
Category		You may assign a category name, if desired.
Description		Your description for the constant.
Is a Constant		Select to enable the constant properties fields.
Value Type	String	Allows the string value to be displayed explicitly.
	Password	Encrypts the value, displaying asterisks (*) in the field.
Value		The value for the constant.

## Project Variables

Variables function as placeholders, having values that are determined when you create a specific Deployment Profile (see [Mapping Variables](#) on page 180). Project variables can be literals or Environmental constants.

As an example, Figure 67 shows a Project variable defined to represent a password of a database user in a target Environment. A system manager assigns an actual value to this variable in the Deployment Profile editor. The value of the assigned Project variable—an Environmental constant—is then used to connect to the database in the target Environment.

**Figure 67** Creating a Project Variable

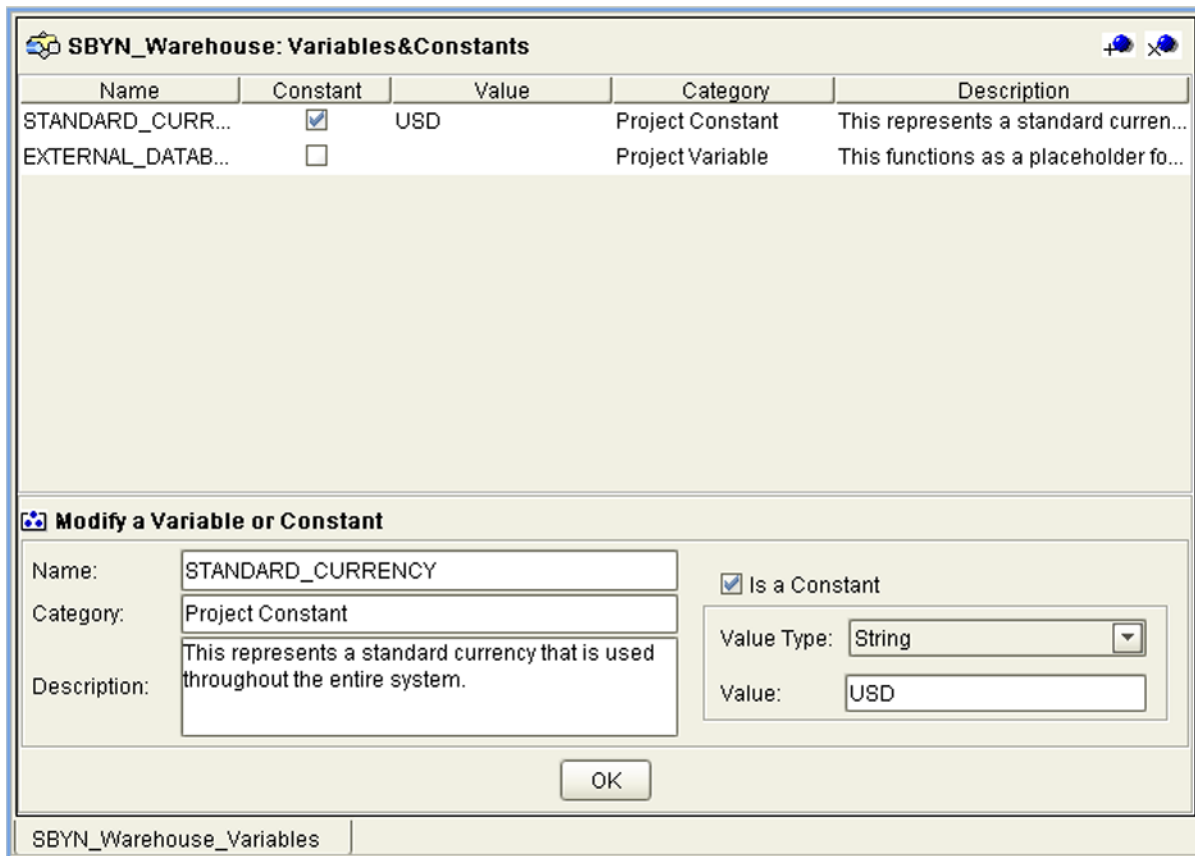
**Table 24** Project Variable Options

Property	Description/Usage
Name	Your name for the Project variable.
Category	You may assign a category name, if desired.
Description	Your description for the variable.
Is a Constant	Does not apply to variables—leave unchecked.
Value Type	Does not apply to variables.
Value	Does not apply to variables.

## Variables & Constants Object Group

Constants and variables are automatically added to a Variables and Constants object group within the Project (see Figure 68). Selecting an entry displays it in the lower panel, where it can be modified. Clicking **OK** overwrites the previous definition with the modified version.

**Figure 68** Variables and Constants Object Group





# Object Type Definitions

This chapter introduces Object Type Definitions (OTDs), describes the features of the OTD Editor, and outlines procedures for creating externally-defined OTDs.

## What's in This Chapter

- [Overview](#) on page 97
- [Using the OTD Wizards](#) on page 98
- [OTD Editor](#) on page 100
- [Creating Externally-Defined OTDs](#) on page 107

---

## 6.1 Overview

The core functionality of eInsight Enterprise Service Bus is handling messages, which it accomplishes by means of Collaborations. To operate on a message, the Collaboration needs a description of the message format. The format description may follow a standard, and be available in some standard metadata format, or it may not be; in the latter case, you need a convenient way to define the format. OTDs provide the solution.

OTDs describe external data formats that characterize the input and output data structures in a Collaboration Definition. OTDs typically have a specific external representation format that is used to store and transport the OTD contents through an ICAN Project. The OTD defines both this external representation and the run-time data structure.

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

As with other eInsight Enterprise Service Bus components, it is essential to manage versions of Object Type Definitions carefully. See [Version Control](#) on page 64 for descriptions of various version control features applicable to Object Type Definitions.

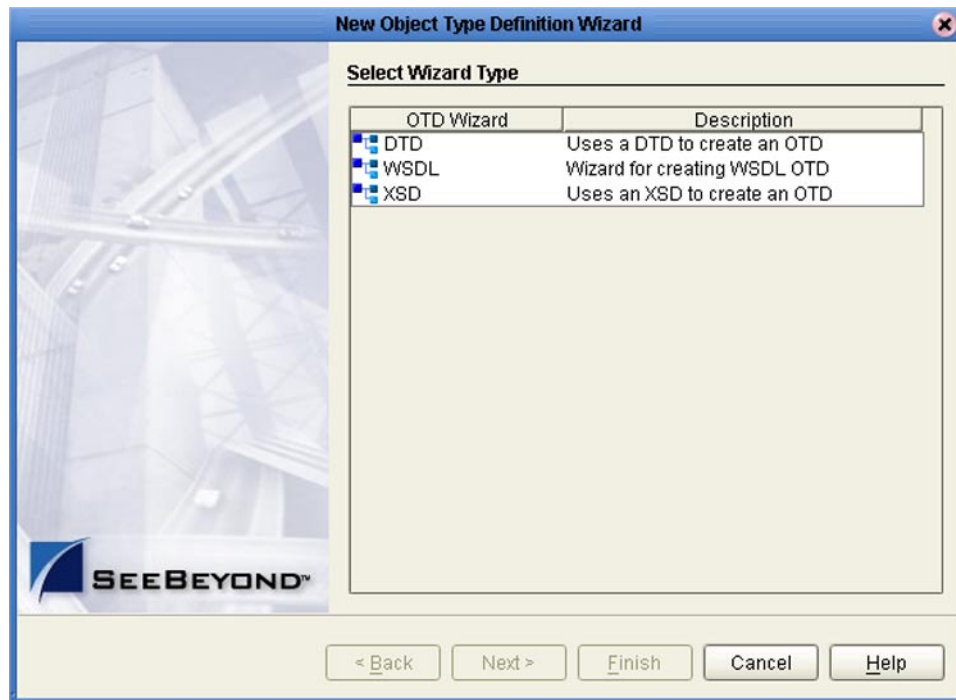
## 6.2 Using the OTD Wizards

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.

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 selection dialog, shown in Figure 69. This initial dialog allows you to select the specific type of OTD Wizard needed for your application. The basic Wizards supplied with eInsight ESB are described in:

- [Using the DTD Wizard](#) on page 107
- [Using the WSDL Wizard](#) on page 112
- [Using the XSD Wizard](#) on page 118

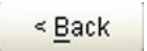

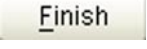
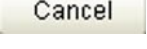

**Figure 69** OTD Wizard Selection Dialog



Additional OTD Wizards are supplied with eInsight Enterprise Service Bus add-on components, 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 69.

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 25 for your reference.

**Table 25** OTD Wizard Navigation Buttons

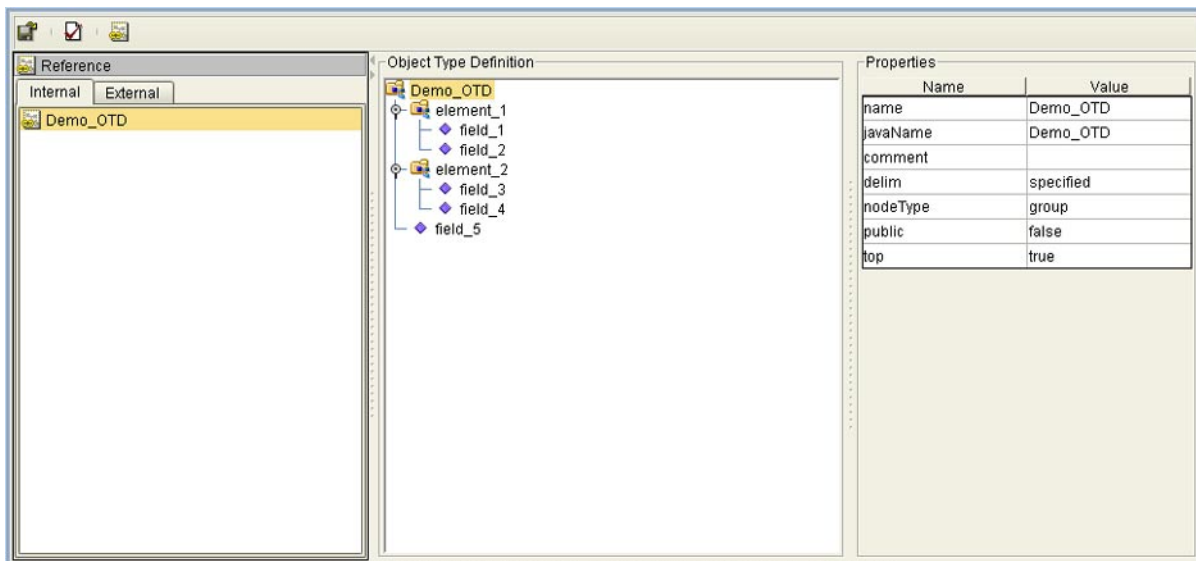
Button	Function
	Returns to the previous step in the wizard. This button is disabled on the first step.
	Goes to the next step in the wizard. This button is disabled on the last step.
	Saves all OTD settings and closes the wizard. This button is only enabled on the last step.
	Closes the wizard without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

## 6.3 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 70. 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.

**Note:** Remember that externally-defined OTDs are read-only, and cannot be edited. You can, however, test them to verify correctness of the build.





**Figure 70** 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.

**Table 26** OTD Editor Toolbar Icons

Icon	Command	Function
	Save as New Name in Repository	Displays a dialog box with which you can save the current OTD under a new name to any Project or Subproject in the Repository. <div style="text-align: center;">  </div>
	Tester	Displays/refreshes the Tester area.
	Toggle Reference Tab Panel	Displays/hides the Reference area.

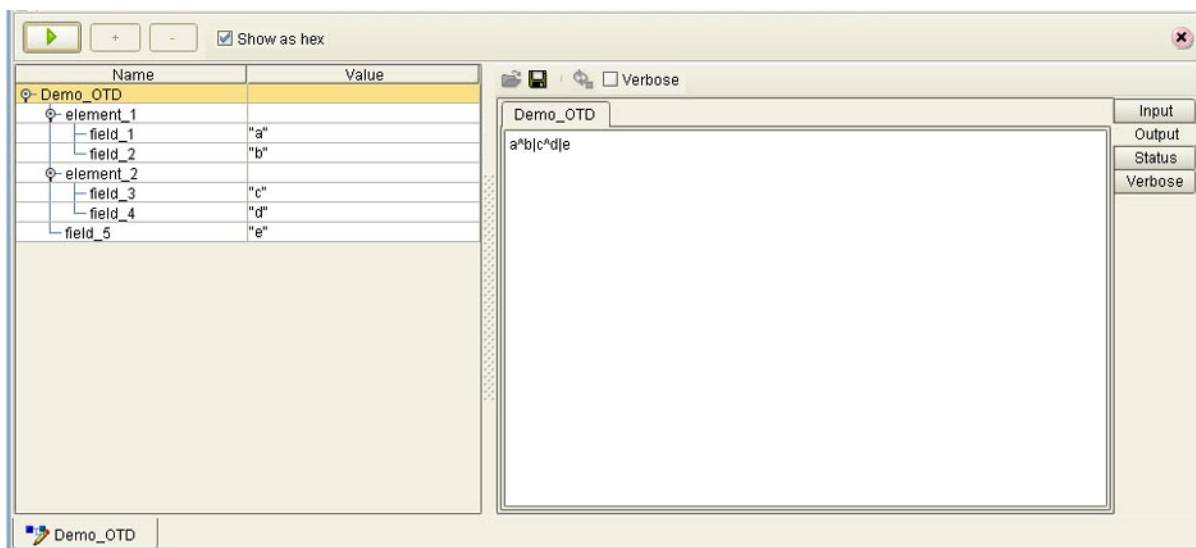
### 6.3.1 OTD Tester

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.

Clicking the Tester icon (see Table 26) in the OTD Editor toolbar saves the OTD to the Repository and displays the OTD Tester as the lower part of the editor, as shown in Figure 71.

**Figure 71** OTD Tester






The data display panel on the right has four data display modes, selectable by tabs:

- Input
- Output
- Status
- Verbose




The **Input** tab is selected by default. Use of the OTD Tester is described in [Using the OTD Tester](#) on page 104.

**Table 27** OTD Tester Buttons

Button	Command	Function
	Run Tester	Runs the tester with the data values entered in the table; the data is marshaled and displayed in the <i>Output</i> panel.
	Add Instance	Adds an instance of an optional node (button is deactivated when an optional instance already exists) or an instance of the selected repeating node. The repeating node is added immediately following the selected instance. You can also select the length field for a repeating node, in which case the new node will be added as the first instance.
	Delete Instance	Deletes the selected instance of an optional or repeating node. Button is deactivated if the optional instance does not exist.

Next to the OTD Tester buttons is a check box labeled **Show as hex**. When checked, the values for **byte[]** nodes are shown in hexadecimal format (default setting). When not checked, the values are shown as regular alphanumeric text.

**Table 28** OTD Tester Icons

Icon	Command	Function
	Open	Displays the <i>Open File</i> dialog box, where you select the file you want to open.
	Save	Displays the <i>Save File</i> dialog box, where you select the location in which you want to save the file.
	Refresh	Repopulates the OTD object elements with the values from the data display panel.

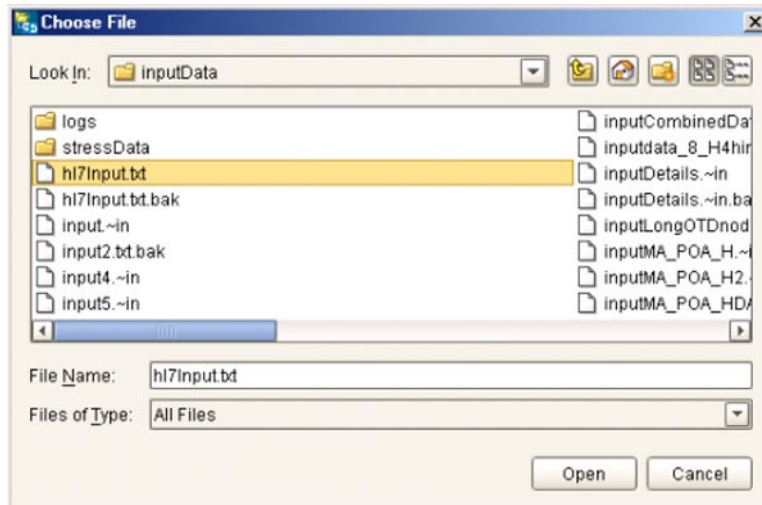
Next to the Collaboration Tester buttons is a check box labeled **Verbose**. Selecting this check box causes parsing information to appear on the *Verbose* panel of the tester.

### 6.3.2 Using the OTD Tester

To test data unmarshaling functionality

- 1 Open or create an OTD.
- 2 Click the **Tester** icon to display the OTD Tester.
- 3 Provide the input test data either by selecting a data file (see Figure 72).

**Figure 72** Select Data File



- 4 Click the **Refresh** button to unmarshal the data.
- 5 Verify the output by checking the values for each element for correctness (see Figure 73).

**Figure 73** 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"

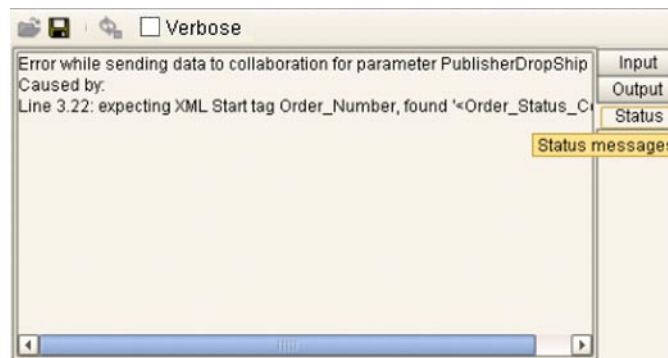


- 6 Save your input test data to a file for re-use by selecting the **Input** data display and clicking the **Save** icon.
- 7 You can also change your test data in the Input data display, then re-test the OTD by clicking the **Refresh** icon (see [Table 27 on page 103](#)) to repopulate your OTD object elements with the new values.

**Note:** Data, which you can edit for testing, is shown in black type.

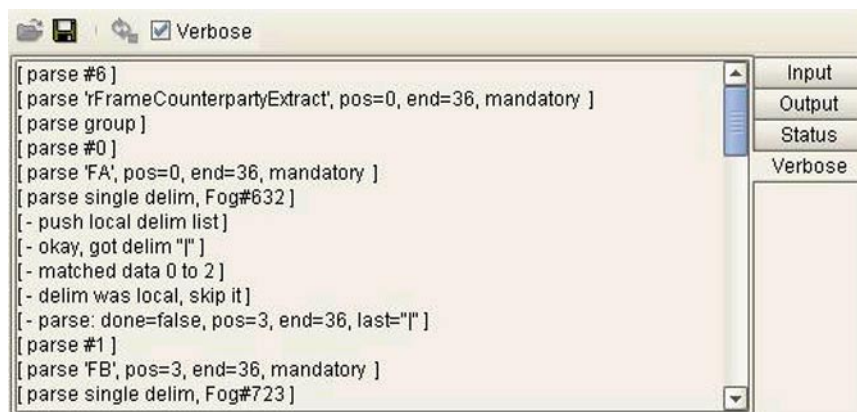
- 8 If there are errors in your input data, the **Status** data display is automatically invoked, showing the appropriate error messages (see Figure 74).

**Figure 74** Status Data Display



- 9 For selected OTDs, the **Verbose** option provides a trace of parsing actions during the *unmarshal* process to aid in debugging the OTD structure. Selecting the **Verbose** check box causes parsing information to appear on the *Verbose* panel (see Figure 75). The format and content of the data display are OTD-specific.

**Figure 75** Verbose Data Display



### To test data marshaling functionality

- 1 Open or create an OTD.
- 2 Click the **Tester** icon to display the OTD Tester.
- 3 Enter or change data values for each node in the **Value** column of the node table (see Figure 76). Use the Add/Delete Instance (+/-) buttons to add or remove instances where appropriate.

**Figure 76** OTD Tester Node Table

Name	Value
⊖ Demo_OTD	
⊖ element_1	
field_1	"a"
field_2	"b"
⊖ element_2	
field_3	"c"
field_4	"d"
field_5	"e"

- 4 Click the **Run Tester** button to marshal (serialize) the data.
- 5 The output is displayed in the *Output* panel (see Figure 77).

**Figure 77** Serialized Data in Output Panel



## 6.4 Creating Externally-Defined OTDs

Externally-defined OTDs are based on formats or standards external to eInsight Enterprise Service Bus, such as Document Type Definition (DTD), Web Services Definition Language (WSDL), and XML Schema Definition (XSD). Some of these OTDs are *messagable*, others are API-based. Externally-defined OTDs are read-only.

### 6.4.1 Using the DTD Wizard

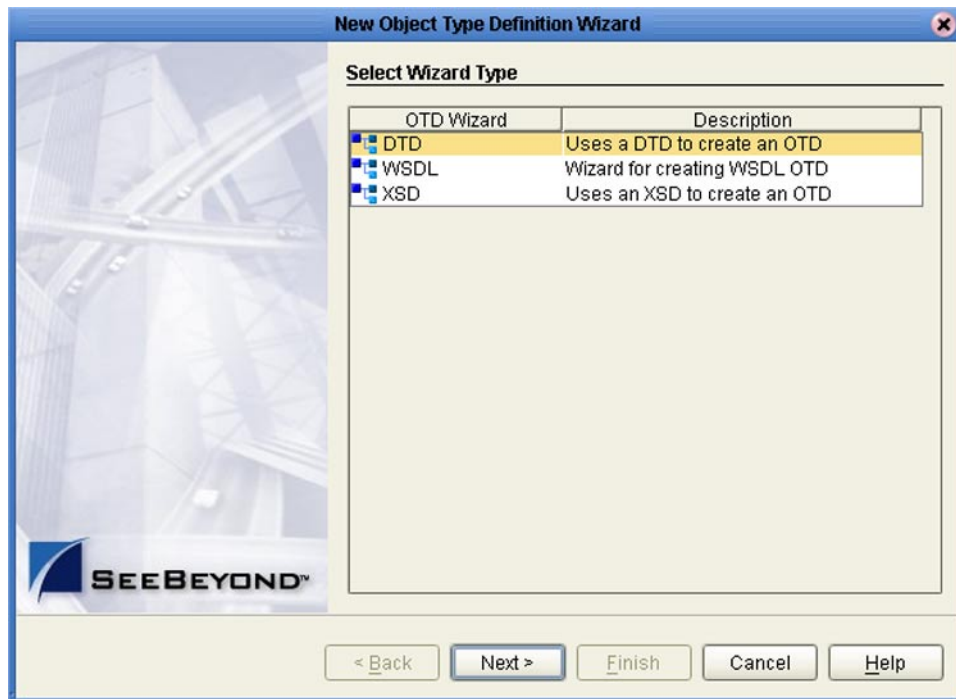
**Note:** A newly-created DTD OTD can be re-defined by selecting the **Relaunch** option from its context menu in Enterprise Explorer (see **Object Type Definition Menu** on page 82). This relaunches the DTD Wizard, and allows you to reselect files and options while preserving the original OID (imported OTDs cannot be modified in this manner).

**Note:** When using the **Relaunch** option, the node structure of the OTD must not change.

To create an OTD file from a DTD file

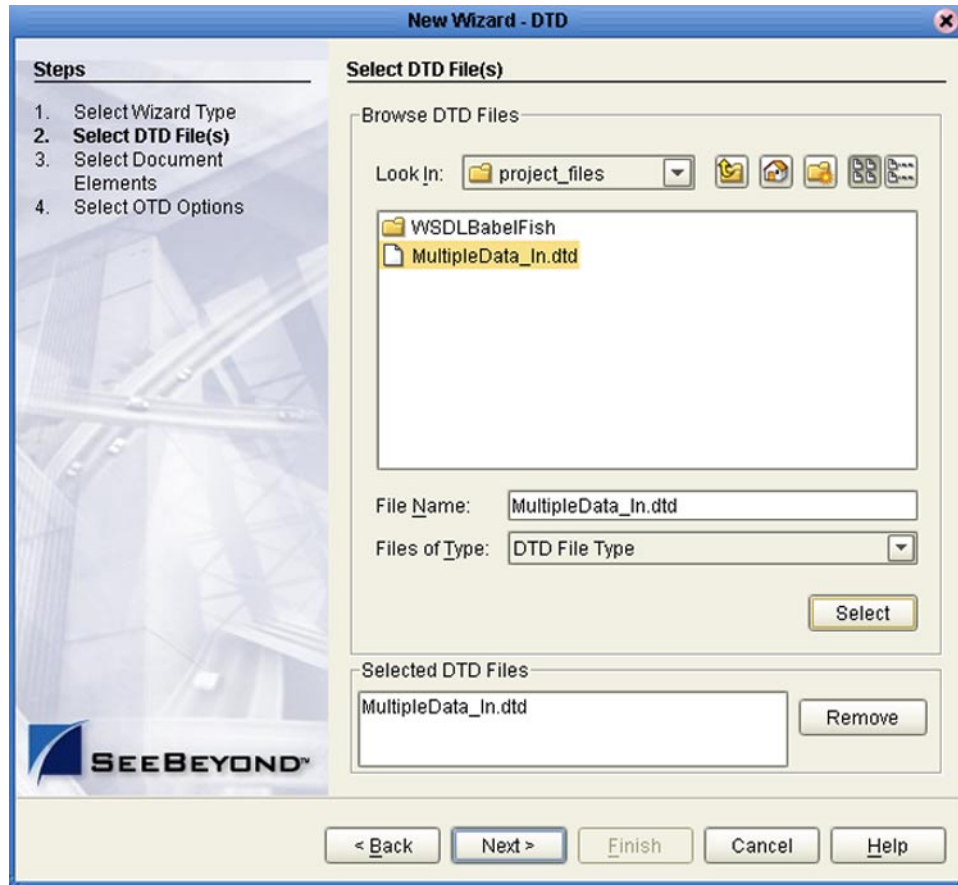
- 1 In the *Select Wizard Type* dialog, select **DTD** from the *OTD Wizard* list (see Figure 78) to create an OTD file from a Document Type Definition (DTD) file.

**Figure 78** OTD Wizard Selection: DTD Wizard



- 2 Click **Next** to display the *Select DTD File(s)* dialog box, shown in Figure 79.

**Figure 79** Select DTD File(s) Dialog Box



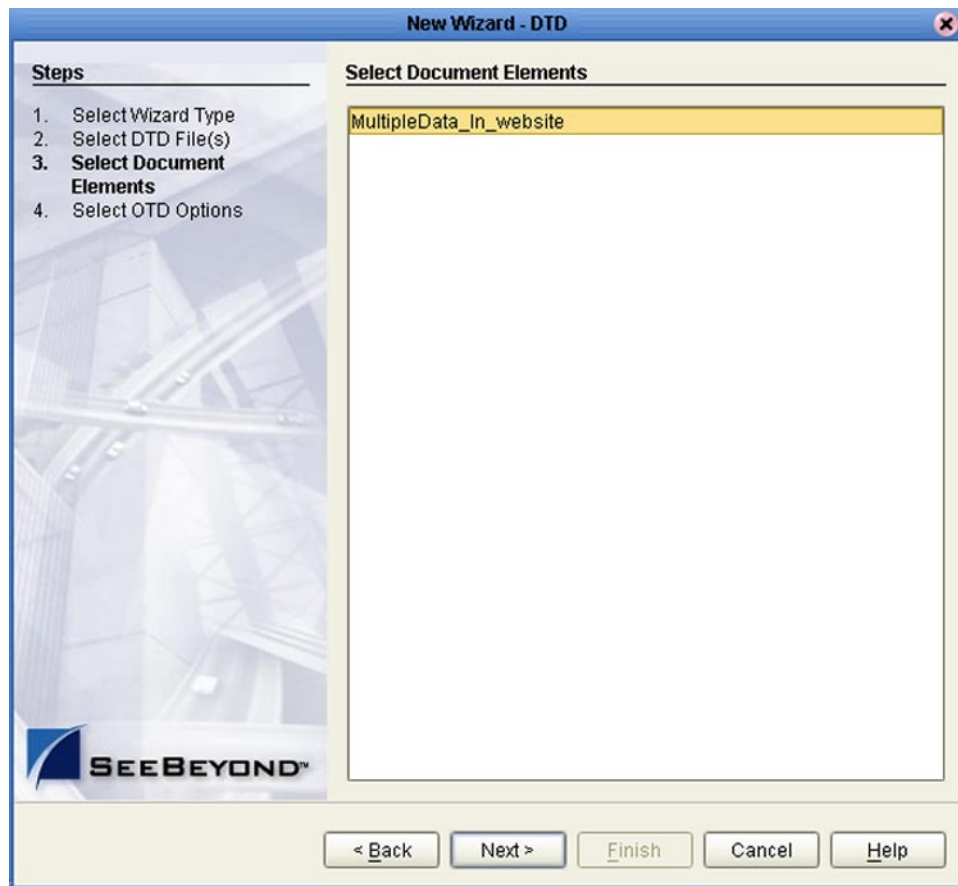
- 3 In the *Look In* drop-down list, navigate to the DTD file or files that you want to use to create the OTD. Click **Select** to add the files to the *List of Selected DTDs*.
  - ◆ If the DTD file does not contain all information required for building an OTD (such as element definitions) a warning box such as that shown in Figure 91 will be displayed.

**Figure 80** Cannot Create OTD Warning Box



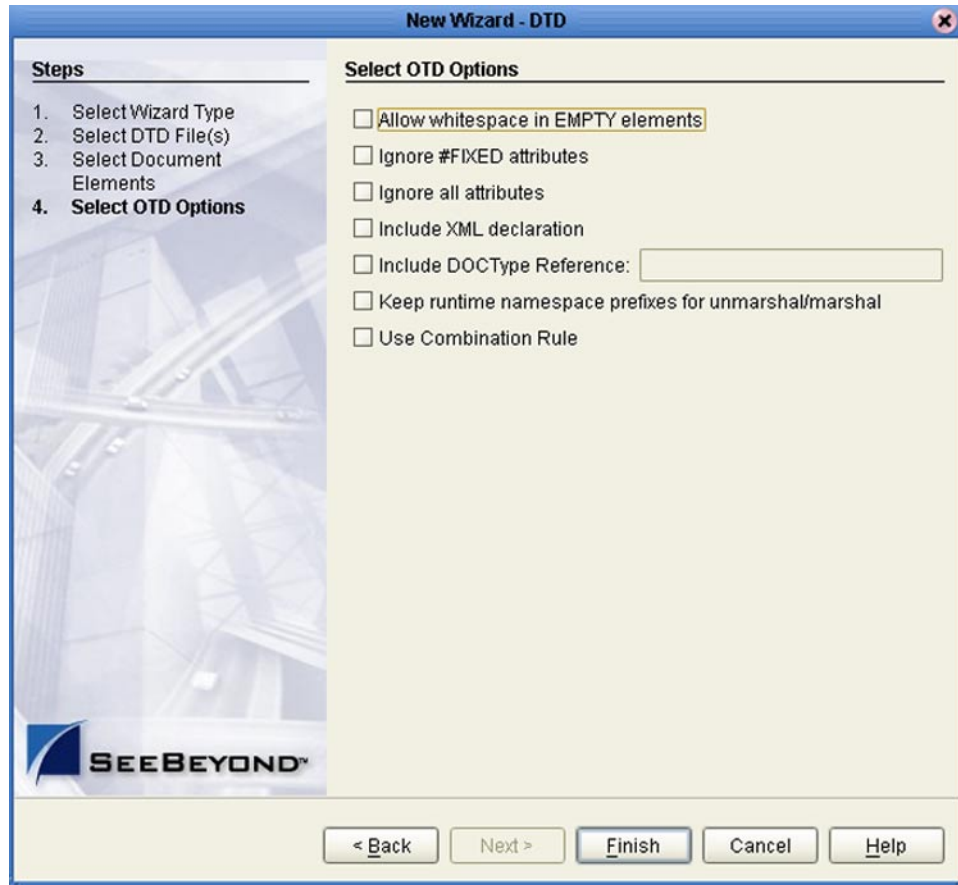
- 4 Click **Next** to display the *Select Document Elements* dialog box, shown in Figure 81.

**Figure 81** 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 82.

**Figure 82** Select OTD Options Dialog Box



7 Select the check boxes next to the OTD options you want to enable (see Table 29).

**Table 29** DTD OTD Options

Option	Description
Allow whitespace in EMPTY elements	If an element is defined as EMPTY, this option controls whether or not white spaces are allowed within the element in the XML instance document.
Ignore #FIXED attributes	<p>This option controls whether or not attributes defined as FIXED are ignored during the unmarshal and marshal processes.</p> <ul style="list-style-type: none"> <li>▪ If this option is <i>not</i> selected, the attribute is recognized and saved into the OTD's runtime structure during the unmarshal process, and also appears in the output during the marshal process.</li> <li>▪ If this option is selected, the attribute is ignored and neither of the above occurs.</li> </ul>

Option	Description
Ignore all attributes	This option controls whether or not all attributes are ignored during the unmarshal and marshal processes. If both this option and the <i>Keep runtime namespace prefixes ...</i> option (below) are selected, only namespace attributes will be handled during the unmarshal process and consequently presented in the output during the marshal process. (The <i>namespace</i> attribute has the form <b>xmlns:XX</b> .)
Include XML declaration	This option controls whether or not the XML declaration <b>&lt;?xml version="1.0" encoding="....." ?&gt;</b> appears in the output during the marshal process.
Include DOC Type Reference	This option controls whether or not the <b>"&lt;!DOCTYPE ..."</b> string appears in the output during the marshal process.
Keep runtime namespace prefixes for unmarshal/marshal	<p>This option controls whether or not the namespace prefixes used during the marshal process are identical to those used in the unmarshal process.</p> <ul style="list-style-type: none"> <li>▪ If this option is selected, all namespace attributes will be preserved once they appear in the XML instance document, and the namespace prefixes used in the marshal process will be exactly as they were presented in the XML document during the unmarshal process.</li> <li>▪ If this option is <i>not</i> selected, then the namespace prefixes used in the marshal process might be different than the ones presented in the XML document during the unmarshal process (for example, the namespace prefixes that are presented in the XSD file might be used).</li> </ul> <p><b>Note:</b> A consequence of selecting this option is that if there is no unmarshal process performed before the marshal process, then there will be no namespace attributes presented in the output (see the comment for the option below).</p>
Use Combination Rule	Not currently used.

8 Click **Finish** to add the OTD to your Project and invoke the OTD Editor.

## 6.4.2 Using the WSDL Wizard

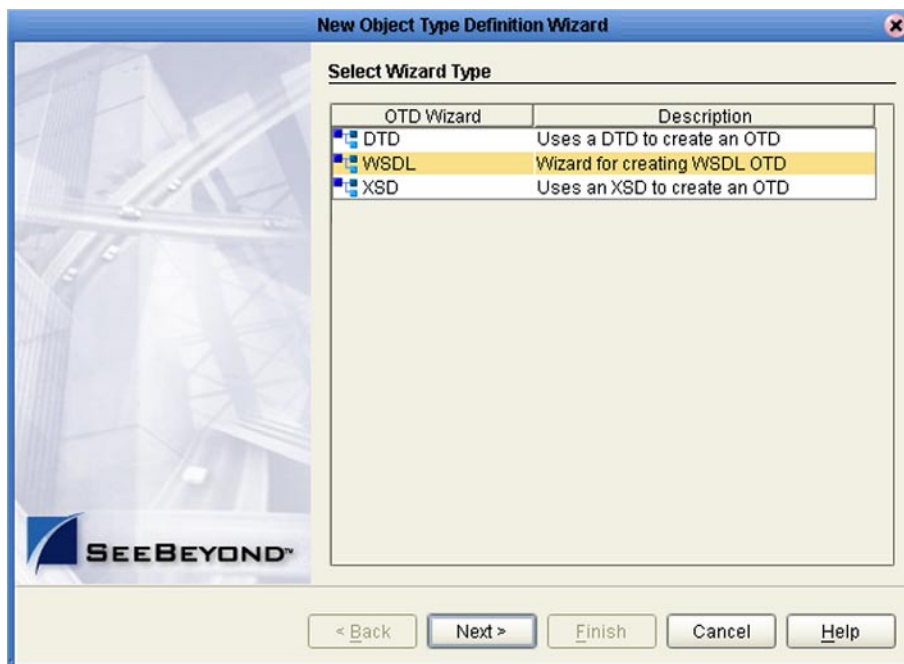
**Note:** A newly-created WSDL OTD can be re-defined by selecting the **Relaunch** option from its context menu in Enterprise Explorer (see **Object Type Definition Menu** on page 82). This relaunches the WSDL Wizard, and allows you to reselect files and options while preserving the original OID (imported OTDs cannot be modified in this manner).

**Note:** When using the **Relaunch** option, the node structure of the OTD must not change.

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 83) to create an OTD file from an WSFL file.

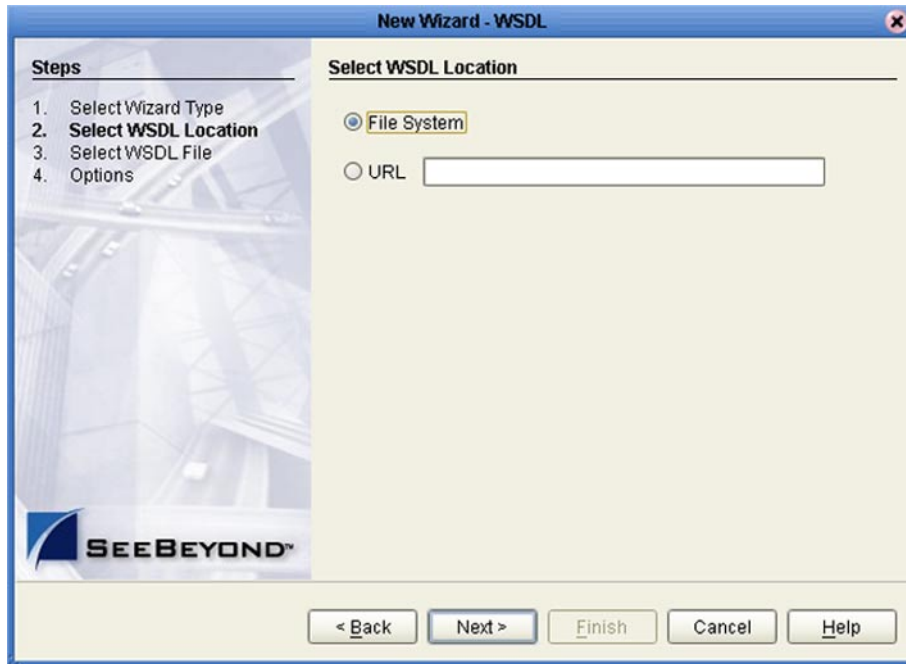
**Figure 83** OTD Wizard Selection: WSDL Wizard



- 2 Click **Next** to display the *Select WSDL File Location* dialog, shown in Figure 84



**Figure 84** WSDL Wizard: Select WSDL Location

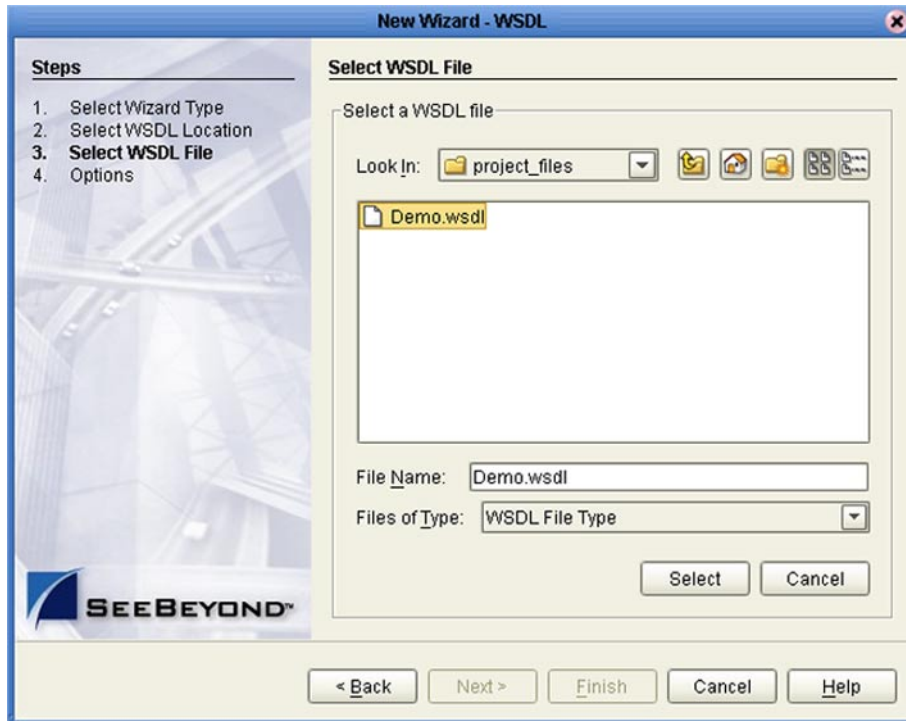


- 3 In the *Select WSDL Location* dialog, select **File System** or enter a **URL**, depending upon where your WSDL file is located.

*Note:* If you are using a URL that points outside your firewall, you must set up a proxy server (see [Using a Proxy Server](#) on page 38).

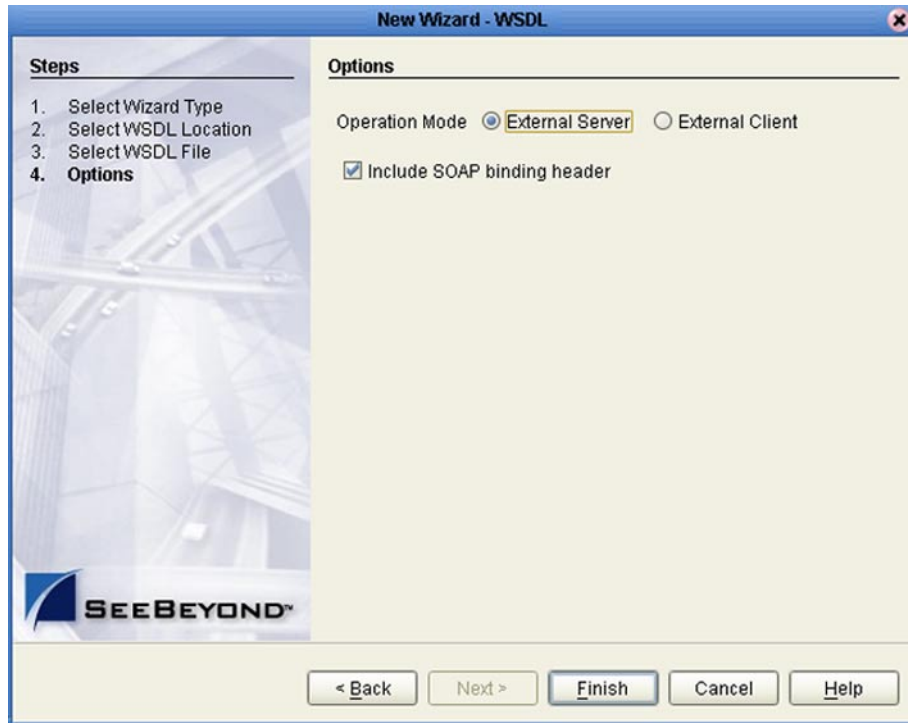
- 4 Click **Next** to display the *Select WSDL File* dialog, shown in Figure 85.

**Figure 85** 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 86.

**Figure 86** WSDL Wizard: Select OTD Options



- 7 Select the required Operational Mode:
  - ◆ If you are building an ICAN Client for an external Web Service, select **External Server**.
  - ◆ If you are implementing and exposing an ICAN Web Service, select **External Client**.
- 8 If you want to include SOAP header elements in the SOAP envelope at runtime (see [SOAP Header Elements](#) on page 117), select the check box for **Include SOAP binding header**.

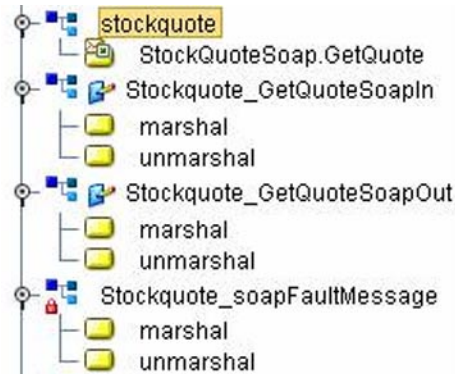
*Note:* To enable this option, the `wsdl:binding` section in the WSDL document selected in Wizard Step 3 must contain the necessary binding information for `soap:header`.

- 9 Click **Finish** to add the OTD to your Project and invoke the OTD Editor.

## WSDL OTD Structure

The WSDL OTD has the following basic structure, as viewed in the Project Explorer:

**Figure 87** Typical WSDL OTD Structure



## 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 identifies messages as being *input* or *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 OTD structure may be depicted as:

```

Root Node
  PortType_XXX
    Operation_XXX
      Input_XXX
      Output_XXX
  
```

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

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.

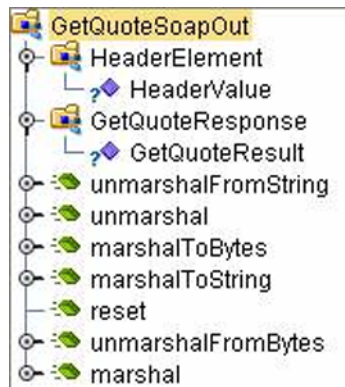
Each operation within a WSDL OTD (like its WSDL file counterpart) uses one of the following operation modes for communication:

- **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 responding message back.

## SOAP Header Elements

When a WSDL OTD is created with the *Include SOAP header* option enabled, the input and output messages include a SOAP header element as shown in Figure 88. If the option is not checked, the header element is missing.

**Figure 88** Output Message with SOAP Header Included



### 6.4.3 Using the XSD Wizard

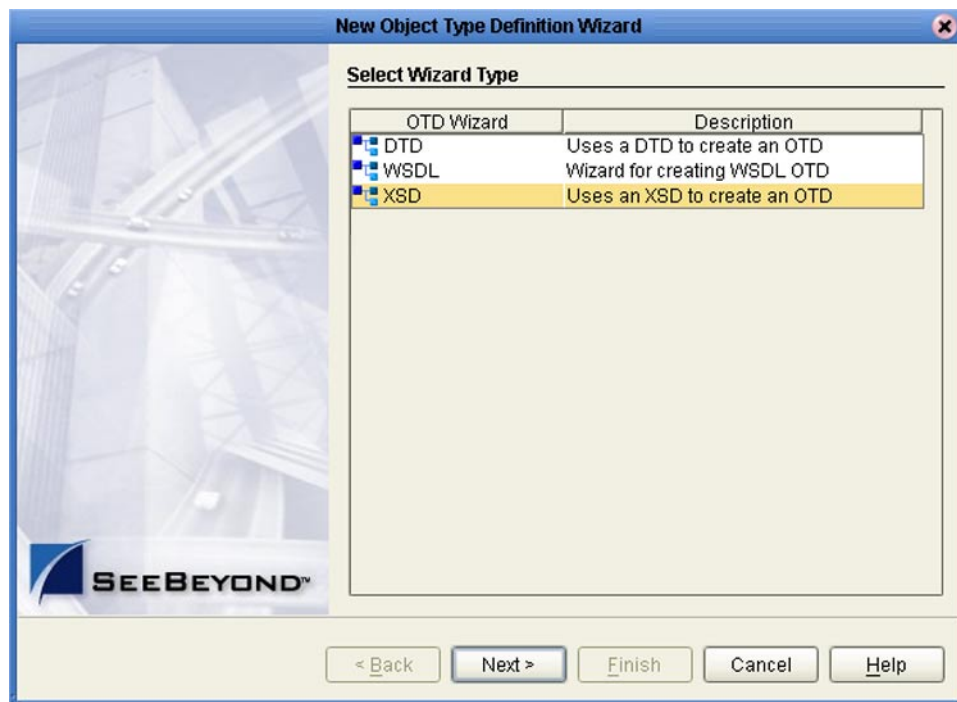
**Note:** A newly-created XSD OTD can be re-defined by selecting the **Relaunch** option from its context menu in Enterprise Explorer (see **Object Type Definition Menu** on page 82). This relaunches the XSD Wizard, and allows you to reselect files and options while preserving the original OID (imported OTDs cannot be modified in this manner).

**Note:** When using the **Relaunch** option, the node structure of the OTD must not change.

To create an OTD file from an XSD file

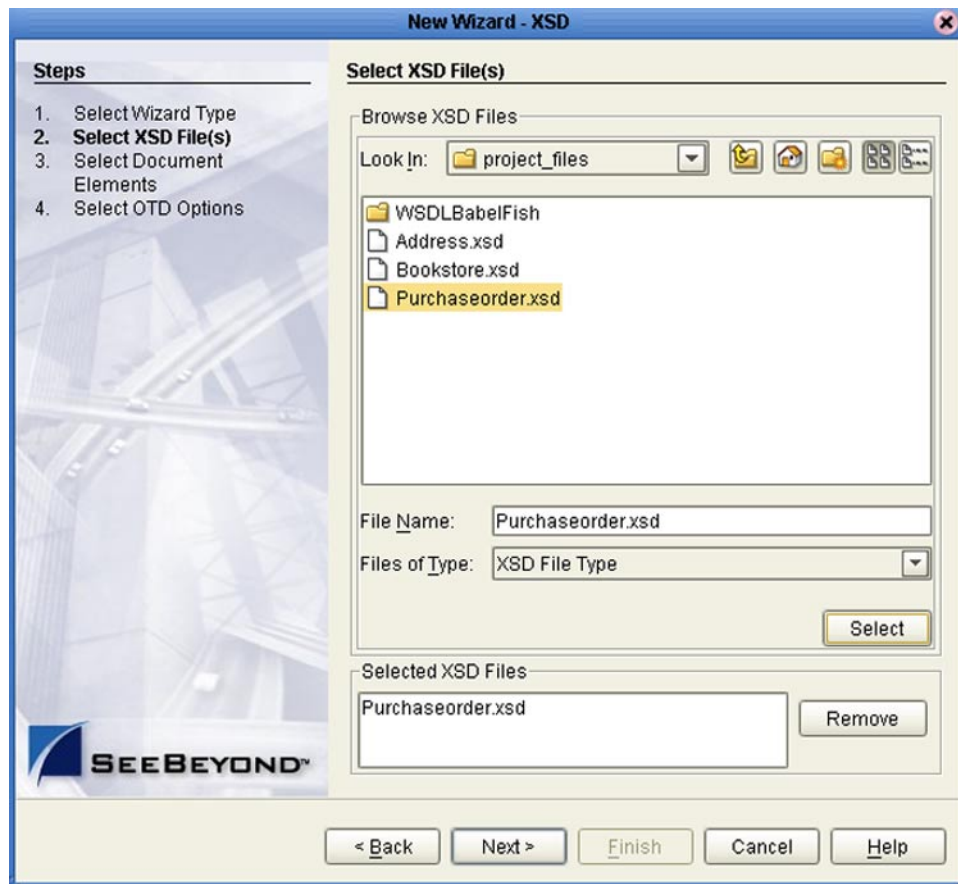
- 1 In the *Select Wizard Type* dialog, select **XSD** from the *OTD Wizard* list (see Figure 89) to create an OTD file from an XSD file.

**Figure 89** OTD Wizard Selection: XSD Wizard



- 2 Click **Next** to display the Select XSD File(s) dialog box, shown in Figure 90.

**Figure 90** XSD Wizard: Select XSD File(s)



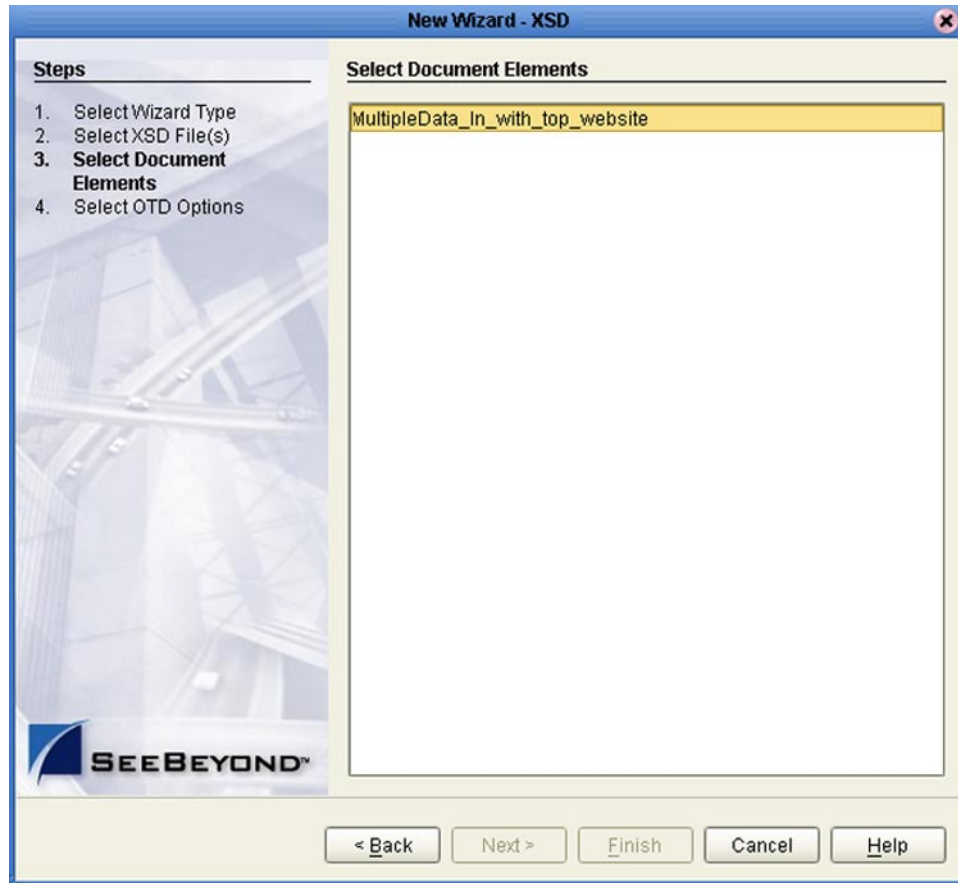
- 3 In the *Look In* drop-down list, navigate to the XSD file or files that you want to use to create the OTD. Click **Select** to add the files to the *List of Selected XSDs*.
  - ◆ If the XSD file does not contain all information required for building an OTD (such as element definitions) a warning box such as that shown in Figure 91 will be displayed.

**Figure 91** Cannot Create OTD Warning Box



- 4 Click **Next** to display the *Select Document Elements* dialog box, shown in Figure 92.

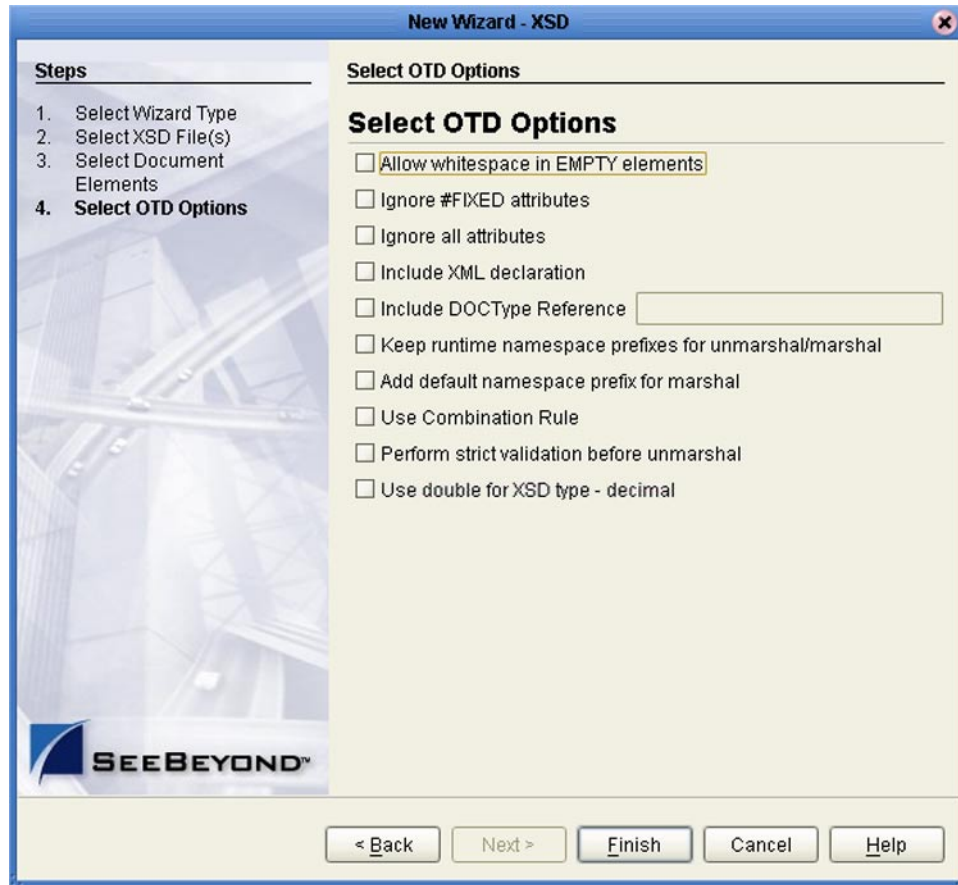
**Figure 92** Select Document Elements Dialog Box



- 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 93.



**Figure 93** Select OTD Options Dialog Box



7 Select the check boxes next to the OTD options you want to enable (see Table 30).

**Table 30** XSD OTD Options

Option	Description
Allow whitespace in EMPTY elements	Not currently used for XSD OTDs.
Ignore #FIXED attributes	<p>This option controls whether or not attributes defined as FIXED are ignored during the unmarshal and marshal processes.</p> <ul style="list-style-type: none"> <li>▪ If this option is <i>not</i> selected, the attribute is recognized and saved into the OTD's runtime structure during the unmarshal process, and also appears in the output during the marshal process.</li> <li>▪ If this option is <i>is</i> selected, the attribute is ignored and neither of the above occurs.</li> </ul>

Option	Description
Ignore all attributes	This option controls whether or not all attributes should be ignored in the unmarshal and marshal processes. If both this option and the <i>Keep runtime namespace prefixes ...</i> option (below) are selected, only namespace attributes will be handled during the unmarshal process and consequently presented in the output during the marshal process. (The <i>namespace</i> attribute has the form <b>xmlns:XX</b> .)
Include XML declaration	This option controls whether or not the XML declaration <b>&lt;?xml version="1.0" encoding="....." ?&gt;</b> appears in the output during the marshal process.
Include DOC Type Reference	Not currently used for XSD OTDs.
Keep runtime namespace prefixes for unmarshal/marshal	<p>This option controls whether or not the namespace prefixes used during the marshal process are identical to those used in the unmarshal process.</p> <ul style="list-style-type: none"> <li>▪ If this option is selected, all namespace attributes will be preserved once they appear in the XML instance document, and the namespace prefixes used in the marshal process will be exactly as they were presented in the XML document during the unmarshal process.</li> <li>▪ If this option is <i>not</i> selected, then the namespace prefixes used in the marshal process might be different than the ones presented in the XML document during the unmarshal process (for example, the namespace prefixes that are presented in the XSD file might be used).</li> </ul> <p><b>Note:</b> A consequence of selecting this option is that if there is no unmarshal process performed before the marshal process, then there will be no namespace attributes presented in the output (see the comment for the option below).</p>
Add default namespace prefix for marshal	<p>This option controls whether or not the prefix of the default target namespace of an element is applied to the element during the marshal process.</p> <ul style="list-style-type: none"> <li>▪ If both this flag and the <i>Keep runtime namespace prefixes ...</i> option (above) are selected, then the default target namespace of an element will be applied to the element during the marshal process, <i>if it is a root element</i>.</li> <li>▪ If the <i>Keep runtime namespace prefixes ...</i> option is <i>not</i> selected, then the elements are qualified based on the XSD definition and this flag has no effect.</li> </ul>
Use Combination Rule	Not currently used.
Perform strict validation before unmarshal	Not currently used.
Use double for XSD type - decimal	If this option is checked, Enterprise Designer will use Java double type for the decimal type in the OTD. If not checked (default) Java BigDecimal type is used, which can handle big decimals with high precision.

8 Click **Finish** to add the OTD to your Project and invoke the OTD Editor.

# Environments

This chapter describes the process of defining eInsight Enterprise Service Bus run-time Environments, and the various components of an Environment.

## What's in This Chapter

- [Overview](#) on page 123
- [Environment Explorer](#) on page 124
- [Environment Editor](#) on page 132
- [Logical Hosts](#) on page 134
- [Integration Servers](#) on page 139
- [Message Servers](#) on page 160

---

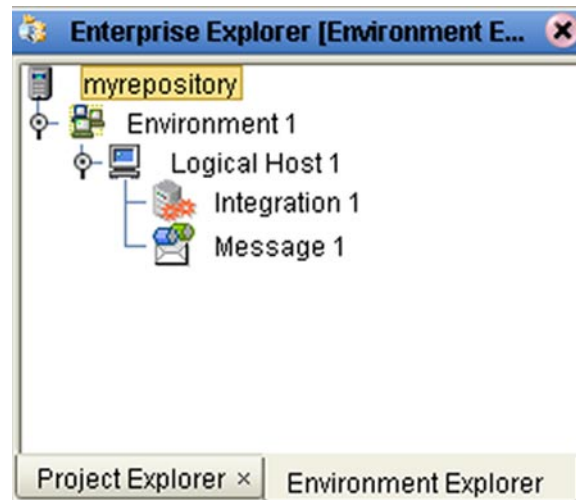
## 7.1 Overview

Projects are run within *Logical Hosts*, which contain the logical resources required by the Project at run time, including integration servers and message servers. The Logical Hosts, in turn, are defined within *Environments*, which represent the physical resources required to implement the Project. The Environment also contains information about external systems with which the ICAN Project interacts.

## 7.2 Environment Explorer

The **Environment Explorer** displays the contents of the Repository that belong to the selected Environment (see Figure 94).

**Figure 94** Enterprise Explorer: Environment Explorer View










The Environment Explorer is used in conjunction with the Environment Editor to create and configure the components of a run-time Environment. Each component in the Environment Explorer has an icon to identify the component type (see [Environment Explorer Icons](#) on page 125). Right-clicking on a component displays a context menu for that component (see [Context Menus](#) on page 126), from which you can select appropriate actions.

## 7.2.1 Environment Explorer Icons

The icons described in Table 31 appear in the Environment Explorer.

**Table 31** Environment Icons

Icon	Function
	Represents the <b>Repository</b> , which is the central ICAN database where all Project information is saved. Binary files required at run time are also stored here.
	Represents a run-time <b>Environment</b> , which contains Logical Hosts and information about external systems.
	Represents a <b>Logical Host</b> , which contains the various logical components and files that are required at run time.
	Represents an <b>Environmental Constant</b> , which you can use to automate eWay and message destination configuration changes.
	Represents a <b>Scheduler</b> component of an Environment, which you can use to set data transfer to occur at set intervals.
	Represents a <b>SeeBeyond Integration Server</b> or third-party <b>application server</b> , which manages the Business Process interactions of an ICAN Project. The integration server is deployed to a Logical Host.
	Represents a <b>SeeBeyond JMS IQ Manager</b> or third-party <b>message server</b> , which is used to store and forward eInsight Enterprise Service Bus system messages. The message server is deployed to a Logical Host.

## 7.2.2 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 126
- [Environment Menu](#) on page 127
- [Logical Host Menu](#) on page 128

### Repository Menu

**Figure 95** Repository Menu

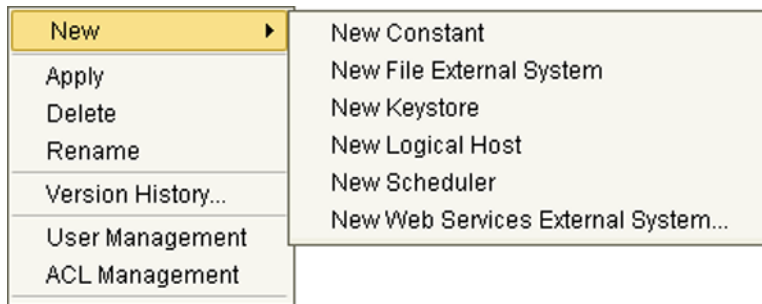


**Table 32** Repository Menu Options

Option	Function
New Environment	Displays a dialog box with which you can create a new Environment.
Configure SNMP Agent	Displays a dialog box with which you can modify the SNMP agent properties.
Save Changes to Repository	Saves any changes you have made in the Environment Editor to the Repository.
Refresh All from Repository	Refreshes the Environment Explorer to display the current contents of the Repository. (Open editors are not refreshed.)

## Environment Menu

**Figure 96** Environment Menu



**Table 33** Environment Menu Options

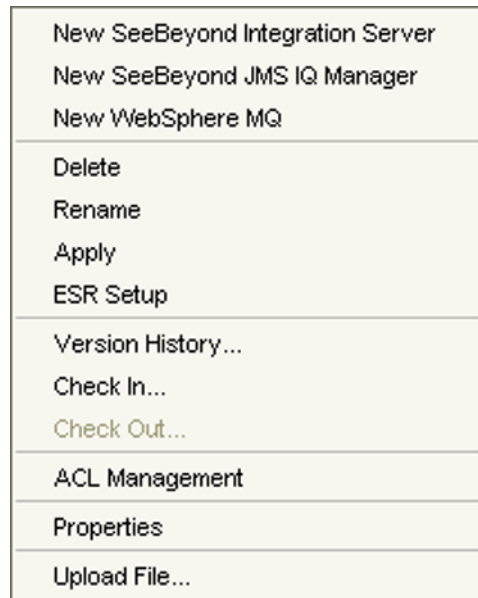
Option		Function
New	Constant	Displays a dialog box with which you can add a constant to the selected Environment. See <a href="#">Defining Environmental Constants</a> on page 132.
	File External System	This option is present only when the File eWay is installed. Other installed eWays also appear as similar options.
	Keystore	Adds a new keystore to the selected Environment.
	Logical Host	Adds a new Logical Host to the selected Environment.
	Scheduler	Displays a dialog box with which you can add a new scheduling component to the selected Environment.
	Web Services External System	Adds a Web Services external system to the selected Environment. See <a href="#">Web Services External System</a> on page 197.
Apply		Applies the latest configuration in the Repository to the selected Environment.
Delete		If you have <i>delete</i> privileges for the Environment (see ACL Management, below), a dialog box is displayed in which you confirm that you want to delete the selected Environment. Clicking <b>Yes</b> then deletes the Environment.
Rename		Activates the field, allowing you to rename the selected Environment.
Version History		Displays a dialog box with which you can track the version history for the selected Environment. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
User Management		Displays a dialog box with which an Administrator can manage message server access. See the <i>eGate Integrator System Administration Guide</i> .

**Table 33** Environment Menu Options

Option	Function
ACL Management	Displays the <b>ACL Properties</b> dialog box, with which an Administrator can assign read/write/delete privileges to users for the selected Environment. See the <i>eGate Integrator System Administration Guide</i> .

## Logical Host Menu

**Figure 97** Logical Host Menu



**Table 34** 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. See the <i>eGate Integrator JMS Reference Guide</i> for details.
New WebSphere MQ	Adds a new IBM WebSphere MQ message server to the selected Logical Host. See the <i>eGate Integrator JMS Reference Guide</i> for details.
Delete	Deletes the selected Logical Host, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the Logical Host (see ACL Management, below).</li> <li>▪ The Logical Host is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected Logical Host. Clicking <b>Yes</b> then deletes the Logical Host.

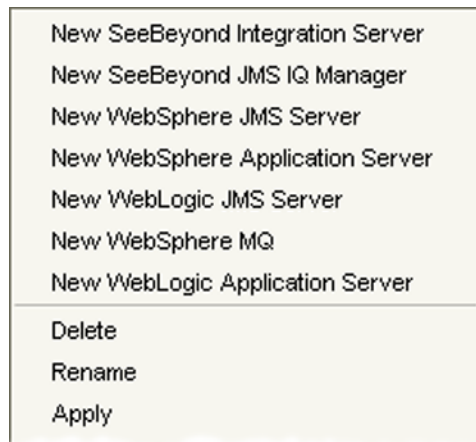


**Table 34** Logical Host Menu Options

Option	Function
Rename	Activates the field, allowing you to rename the selected Logical Host.
Apply	Applies changes to the selected Logical Host.
ESR Setup	Displays a dialog box with which an Administrator can select emergency software releases (ESRs) to add to the Logical Host.
Version History	Displays a dialog box with which you can track the version history for Logical Hosts. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Check In	Displays a dialog box, with which you can check in a new version of an Logical Host. Refer to <a href="#">Checking a Component In</a> on page 66 for more details.
Check Out	Displays a dialog box with which you can check out the current version of an Logical Host. See <a href="#">Checking a Component Out</a> on page 68 for more information.
ACL Management	Displays the <b>ACL Properties</b> dialog box, with which an Administrator can assign read and/or write privileges to users for the selected Logical Host See the <i>eGate Integrator System Administration Guide</i> .
Properties	Displays a dialog box with which you can modify the configuration properties for the selected Logical Host.
Upload File	Allows you to upload third-party libraries (.jar files) to the Logical Host.

**Note:** *If you are using BEA WebLogic and/or IBM WebSphere, the Application Servers and JMS Message Servers for these products will also appear in the context menu (see Figure 98).*

**Figure 98** Logical Host Menu with Third-Party Servers



## Integration Server

The menu shown in Figure 99 is used for both the SeeBeyond Integration Server and third-party application servers. The configuration properties, however, are different.

**Figure 99** Integration Server Menu



**Table 35** Integration Server Menu Options

Option	Function
Version History	Displays a dialog box with which you can track the version history for the selected integration server. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
Delete	Deletes the selected integration server, subject to the following conditions: <ul style="list-style-type: none"> <li>▪ You have <i>delete</i> privileges for the integration server (see ACL Management, below).</li> <li>▪ The integration server is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected integration server. Clicking <b>Yes</b> then deletes the integration server.
Rename	Activates the field, allowing you to rename the selected integration server.
Properties	Displays a dialog box with which you can modify the configuration properties for the selected integration server.
Business Process Debugger	Displays the Business Process Debugger. See the <i>eInsight Enterprise Service Bus User's Guide</i> for information.
ACL Management	Displays the <b>ACL Properties</b> dialog box, with which an administrator can assign read/write/delete privileges to users for the selected integration server. See the <i>eGate Integrator System Administration Guide</i> .

## SeeBeyond JMS IQ Manager

The menu shown in Figure 100 is used for both the SeeBeyond JMS IQ Manager and third-party message servers. The configuration properties, however, are different.

**Figure 100** JMS IQ Manager Menu



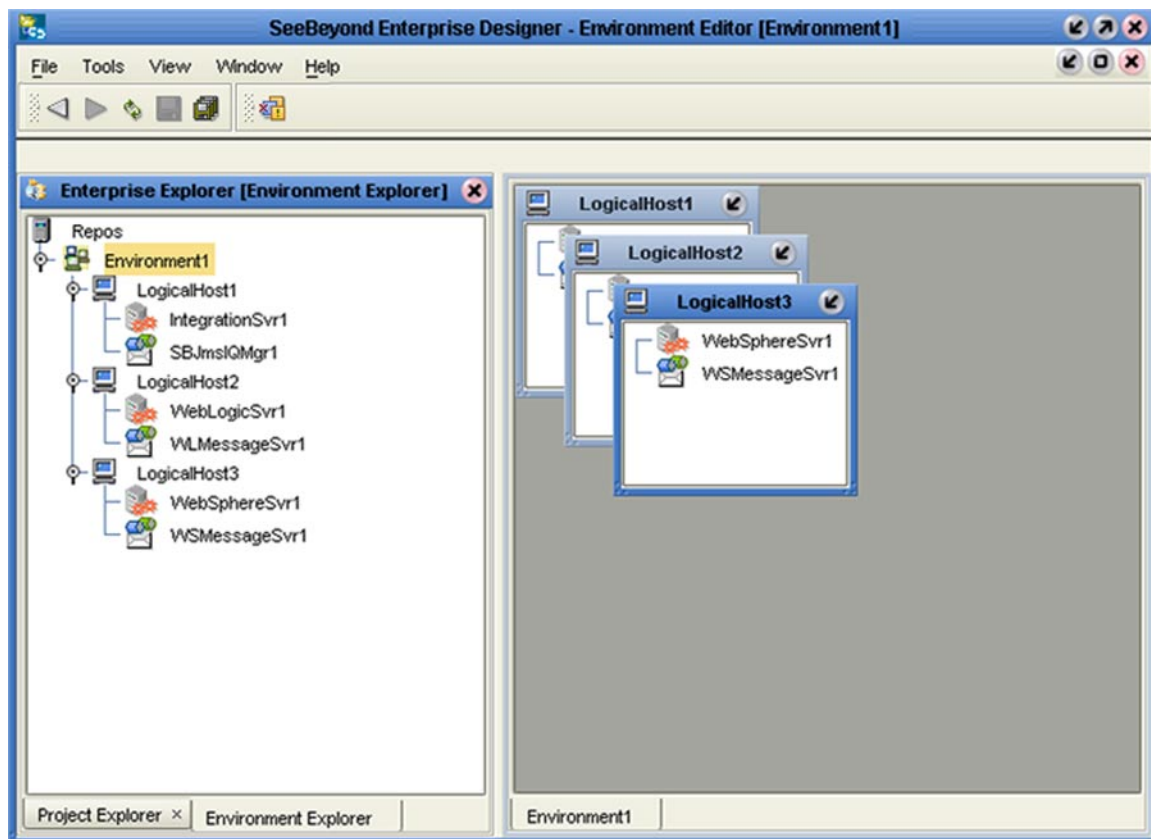
**Table 36** Integration Server Menu Options

Option	Function
Delete	Deletes the selected message server, subject to the following conditions: <ul style="list-style-type: none"> <li>You have <i>delete</i> privileges for the message server (see ACL Management, below).</li> <li>The message server is not checked out by anyone other than yourself.</li> </ul> If these conditions are true, a dialog box is displayed in which you confirm that you want to delete the selected message server. Clicking <b>Yes</b> then deletes the message server.
Rename	Activates the field, allowing you to rename the selected message server.
Properties	Displays a dialog box with which you can modify the configuration properties for the selected message server.
Version History	Displays a dialog box with which you can track the version history for the selected message server. See <a href="#">Viewing a Component's Version History</a> on page 65 for more information.
ACL Management	Displays the <b>ACL Properties</b> dialog box, with which an administrator can assign read/write/delete privileges to users for the selected message server. See the <i>eGate Integrator System Administration Guide</i> .

## 7.3 Environment Editor

Clicking 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 101).

Figure 101 Environment Editor

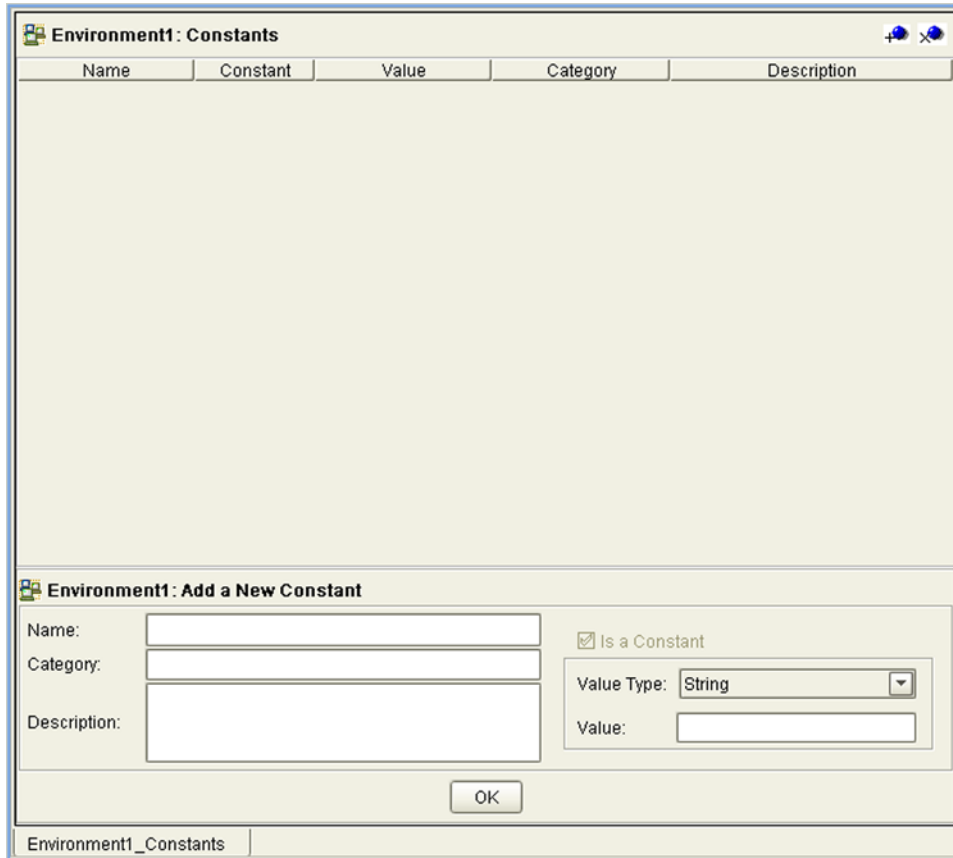


Here you can see the various components (Logical Hosts, servers, and external systems) included in the selected Environment. New Environments are added through the use of the Repository context menu (see [Repository Menu](#) on page 126). Components are added to the Environment by selecting options in the Environment and Logical Host context menus (see [Environment Menu](#) on page 127 and [Logical Host Menu](#) on page 128, respectively).

### 7.3.1 Defining Environmental Constants

Environmental constants are name/value pairs that are visible across the Environment. Selecting the **New Constant** option from the Environment context menu displays the Constants panel in the Environment Editor (see Figure 102).



**Figure 102** Environmental Constants Panel



All constants defined for the specific Environment are listed in the *Constants* section of the panel, along with their various properties. New constants are added using the *Add a New Constant* section of the panel.

**Note:** *When you create an Environmental constant, you assign a permanent value to it—which cannot be overridden.*

**Table 37** Environmental Constants Panel Icons

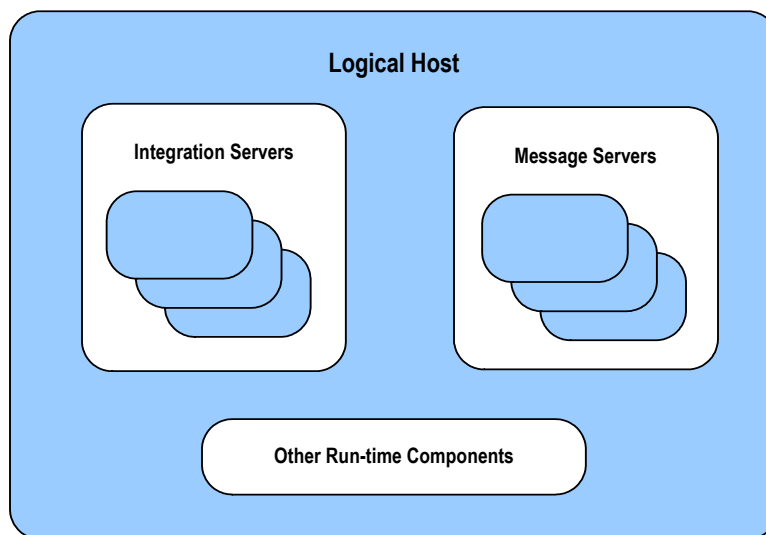
Icon	Name	Function
	Add a New Constant	Adds a new constant to the list.
	Delete a Highlighted Constant	Deletes the selected constant from the list.

## 7.4 Logical Hosts

### 7.4.1 Overview

A Logical Host contains the eInsight Enterprise Service Bus run-time components that are installed on a host hardware platform. A Logical Host can be a member of only one Environment, but each Environment can contain multiple Logical Hosts. Both integration servers and message servers are deployed to the Logical Host, as illustrated in Figure 103.

**Figure 103** Logical Hosts



The master service of the Logical host is the Management Agent. This 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 message servers and integration servers. You can view the status of these components by means of the Enterprise Monitor.

The Enterprise Monitor also allows you to control the components within the Logical Host. Stopping the Logical Host through the Enterprise Monitor stops all components except the Management Agent, which remains running so that the components can be restarted from the Enterprise Monitor. This behavior is necessary to accommodate situations where the Logical Host and Repository are located on separate computers. See the *eGate Integrator System Administration Guide* for additional information.

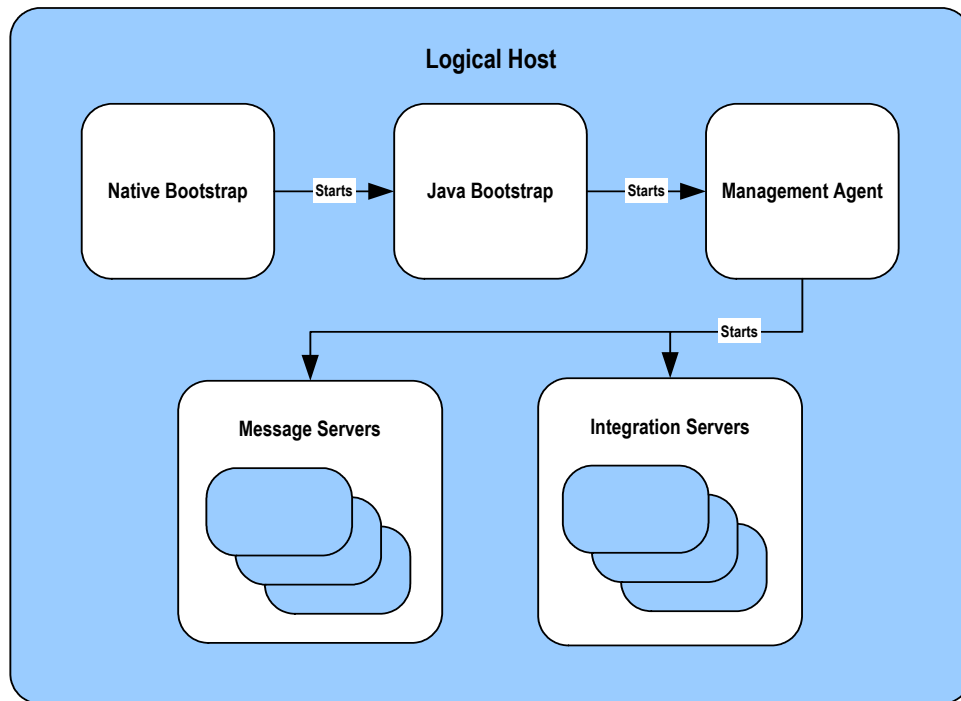
## 7.4.2 Bootstrapping

At run time, a platform-specific bootstrap script starts a Java bootstrap program. The first time the bootstrap runs, it downloads the Management Agent, message server, and integration server from the Repository. The Management Agent is then started, which in turn starts the message server(s) and integration server(s). Figure 104 illustrates this sequence.

Subsequent bootstraps simply start the existing Management Agent without downloading component configurations from the Repository, unless you force a download by:

- Selecting **Apply** from the Logical Host context menu (see [Logical Host Menu](#) on page 128).
- Using the command-line option (-f). See the *eGate Integrator System Administration Guide* for information.

**Figure 104** Startup Sequence



Each Logical Host has a separate bootstrap process. The process is started from a script (*ICAN-root*\logicalhost\bootstrap\bin\bootstrap.bat) or (*ICAN-root*/logicalhost/bootstrap/bin/bootstrap.sh). The Repository location is furnished either by command-line parameters or the configuration properties file (*ICAN-root*\logicalhost\bootstrap\config\logical-host.properties). See the *eGate Integrator System Administration Guide* for additional information.

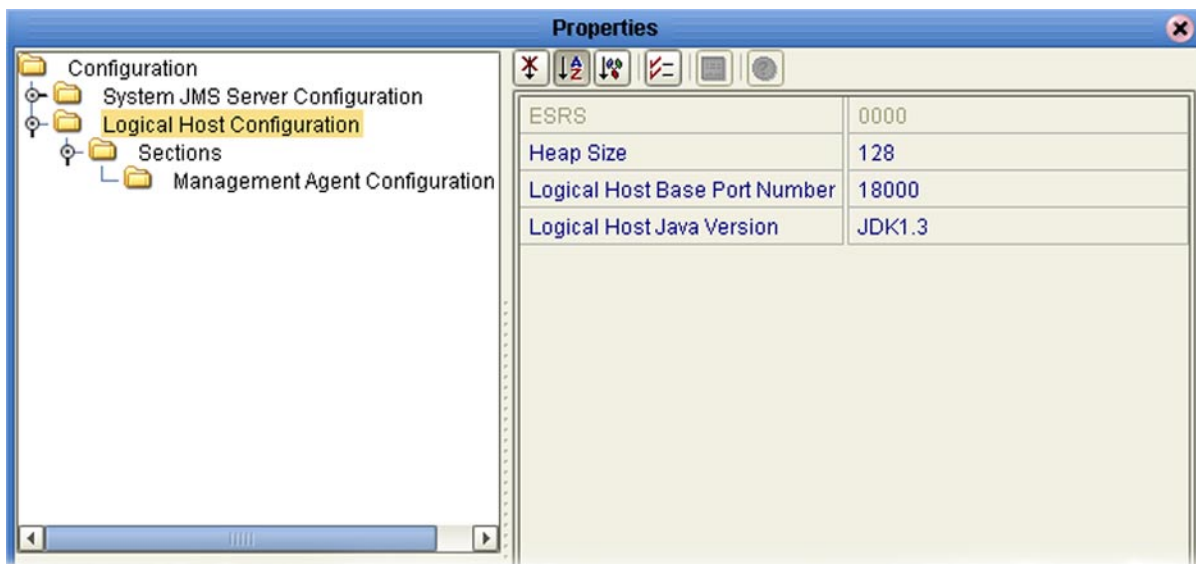
### 7.4.3 Configuring a Logical Host

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

To access the configuration properties for a Logical Host

- 1 Right-click a Logical Host in the Environment Explorer tree to display the context menu for that Logical Host instance.
- 2 Select **Properties** from the context menu to display the **Properties** dialog box.
- 3 Select the **Logical Host Configuration** node in the properties tree to display the Logical Host Configuration Section, which contains the top-level configuration properties for the Logical Host (see Figure 105).

**Figure 105** Logical Host Configuration Properties



**Table 38** Logical Host Configuration Properties List

Property	Description
ESRS	Shows a list of all installed Emergency Software Releases (ESRs).
Heap Size	Specifies the Heap size in Megabytes; the minimum size is <b>128 Mb</b> , which is the default value. Note that this property is only for the bootstrap and management processes, and does not affect the integration server or any runtime components that are processing data.
Logical Host Base Port Number	Specifies the base port number for the Logical Host. The default value is <b>18000</b> . When multiple Logical Hosts reside on a single hardware platform, you must configure the base port numbers; see the following section.



**Table 38** Logical Host Configuration Properties List

Property	Description
Logical Host Java Version	Specifies the Java version being used to the eWay RAR file generation program, so that any generated file will be properly compatible. The default value is <b>JDK1.3</b> .

## Configuring the Base Port Number

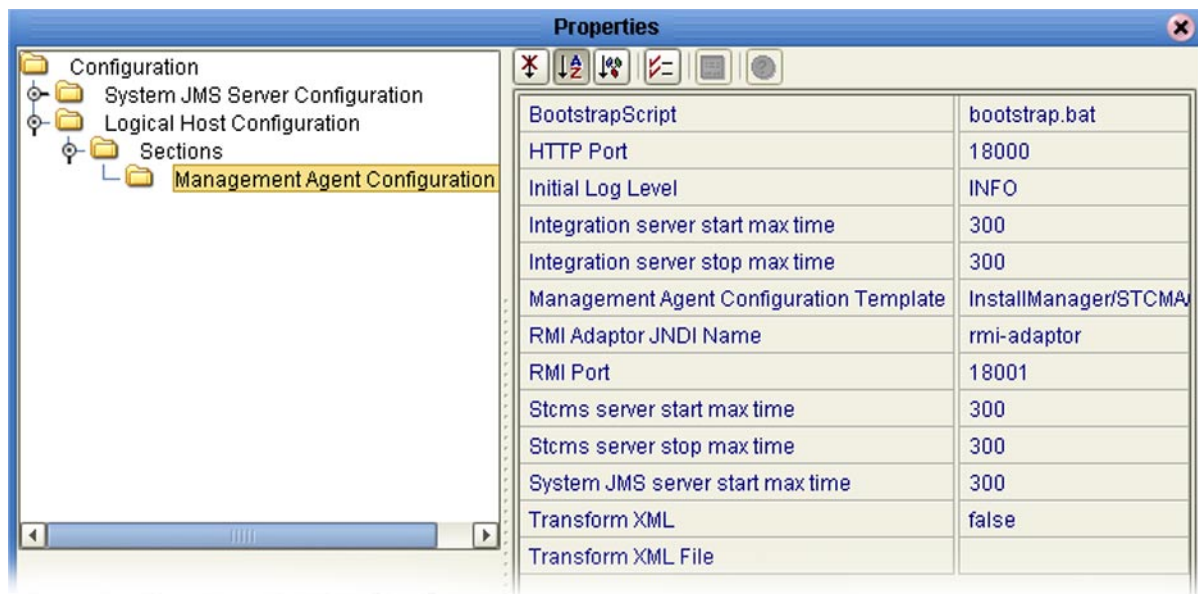
If multiple Logical Hosts concurrently run on the same computer, you must ensure that each Logical Host has a different base port number to avoid conflicts. This base port number is propagated throughout the Logical Host, so that the various components are automatically given successive port numbers following the number 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*).

*Note:* While Windows accepts port numbers below 12000, UNIX does not.

**Figure 106** Management Agent Configuration Properties



**Table 39** Management Agent Configuration Properties List

Property	Description
BootstrapScript	Name of the bootstrap script; the default is <b>bootstrap.bat</b> .
HTTP Port	The HTTP port; the default value is <b>18000</b> .
Initial Log Level	The initial log level (OFF, ALL, DEBUG, INFO, WARN, ERROR, FATAL); the default value is <b>INFO</b> . See the <i>eGate Integrator System Administration Guide</i> for log level details.
Integration server start max time	The maximum time period (in seconds) during which the Management Agent will attempt to start an Integration Server. The default value is <b>300</b> seconds.
Integration server stop max time	The maximum time period (in seconds) during which the Management Agent will attempt to stop an Integration Server. The default value is <b>300</b> seconds.
Management Agent Configuration Template	The name and path of the configuration template; the default value is <b>InstallManager/STCMA/common/config/templates/ManagementAgent-config.vm</b> .
RMI Adaptor JNDI Name	The Java Naming and Directory Interface (JNDI) name of the Remote Method Invocation (RMI) adaptor; the default value is <b>rmi-adaptor</b> .
RMI Port	The Remote Method Invocation (RMI) port. The displayed value depends upon the number of the Logical Host base port; the default value is <b>18001</b> .
Stcms server start max time	The maximum amount of time (in seconds) that the Management Agent will attempt to start a SeeBeyond JMS IQ Manager. The default value is <b>300</b> seconds.
Stcms server stop max time	The maximum amount of time (in seconds) that the Management Agent will attempt to stop a SeeBeyond JMS IQ Manager. The default value is <b>300</b> seconds.
System JMS server start max time	The maximum amount of time (in seconds) that the Management Agent will attempt to start the message server used internally by the ICAN Suite. The default value is <b>300</b> seconds.
Transform XML	The default value is <b>false</b> , and should not be changed.
Transform XML File	This property is not used.

See the *eGate Integrator System Administration Guide* for additional information.

## 7.5 Integration Servers

The Logical Host contains one or more instances of a J2EE-compatible integration server. This server is the engine that runs Collaborations for processing business logic, and eWays for communicating with external applications. The integration server provides services for security, transactions, business rules execution, and connectivity management. eInsight Enterprise Service Bus contains the SeeBeyond Integration Server, and also supports the use of third-party application servers such as BEA WebLogic and IBM WebSphere (see [Deploying Projects to Third-Party Servers](#) on page 185).

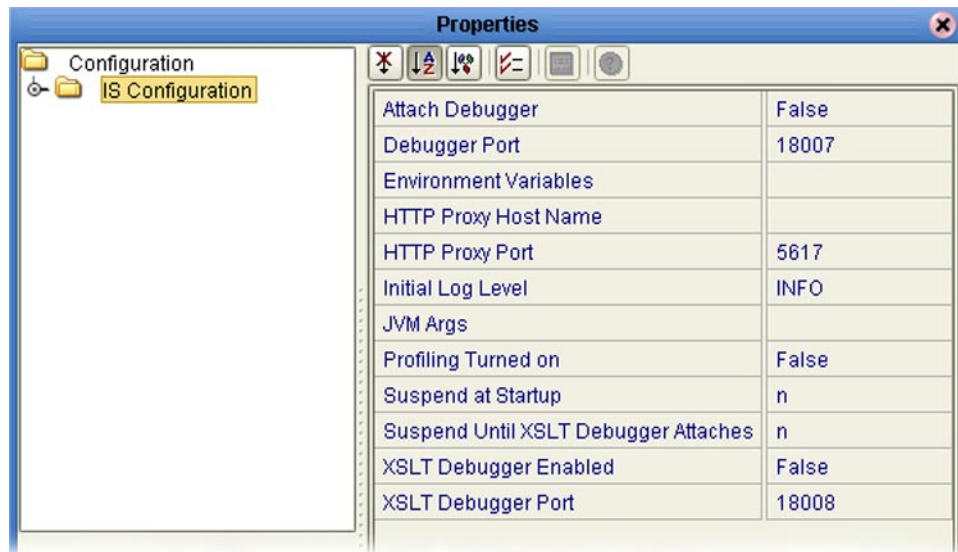
### 7.5.1 Configuring an Integration Server

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

To access the configuration properties for an integration server

- 1 Right-click an integration server in the Environment Explorer tree to display the context menu for that instance.
- 2 Select **Properties** from the context menu to display the **Properties** dialog box.
- 3 Select the **IS Configuration** node in the properties tree to display the top-level IS configuration properties (see Figure 107).

**Figure 107** Top-level IS Configuration Properties



**Table 40** Top-level IS Configuration Properties List

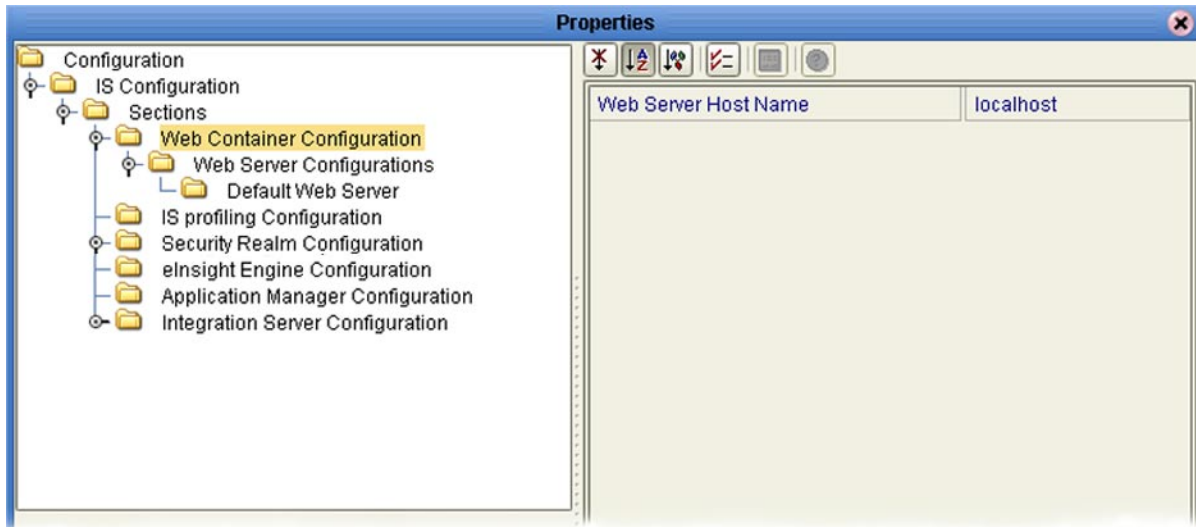
Property	Description
Attach Debugger	Enables/disables debugging for the IS. The default is <b>False</b> (disabled).
Debugger Port	This property is used only when the Debugger is enabled. The port numbers are dynamically assigned; the number shown is typical.
Environment Variables	Specifies user-defined Environment Variables. Entries here override the system settings—so if <i>any</i> variables are specified, then <i>all</i> required variables must be specified. Each element has the format <b>name=value</b> . There is no default.
HTTP Proxy Host Name	Name of the HTTP proxy host, if any. There is no default. Note: Do not use underscore character in the host name.
HTTP Proxy Port	Port for the HTTP proxy host, if any. The default is <b>5617</b> .
Initial Log Level	The initial log level (OFF, ALL, DEBUG, INFO, WARN, ERROR, FATAL); the default value is <b>INFO</b> . See the <i>eGate Integrator System Administration Guide</i> for log level details.
JVM Args	Java Virtual Machine (JVM) arguments. Each element in the collection should specify one, and only one, argument.
Profiling Turned On	Enables/disables performance monitoring for the IS. The default is <b>False</b> (disabled). To enable, change to <b>True</b> (as shown) and configure the properties described in <a href="#">Performance Monitoring (Profiling) Configuration</a> on page 144.
Suspend at Startup	Allows the VM to begin executing before the debugger application attaches. The default is <b>n</b> (do not suspend).
Suspend Until XSLT Debugger Attaches	Allows the VM to begin executing before the XSLT debugger application attaches. The default is <b>n</b> (do not suspend).
XSLT Debugger Enabled	Enables/disables XSLT debugging for the IS. The default is <b>False</b> (disabled).
XSLT Debugger Port	This property is used only when the XSLT Debugger is enabled. The port numbers are dynamically assigned; the number shown is typical.

The IS Configuration node contains several sections, each containing detailed configuration properties for a particular IS component (including the integration server itself). You can also access these properties by selecting **Properties** from the context menus for the appropriate nodes.

## Web Container Configuration

Properties included here are used for setting up Web Services.

**Figure 108** Web Container Configuration Properties



**Table 41** Web Container Configuration Properties List

Property	Description
Web Server Host Name	Specifies the host name; the default is <b>localhost</b> . Note: Do not use underscore character in the host name.

## Web Server Configurations

This directory serves as a container for the configuration properties for all Web servers associated with the selected integration server. If all Web servers in the IS share common properties, these properties can be set in the *Default Web Server*. Otherwise, you should create separate Web Servers as described below.

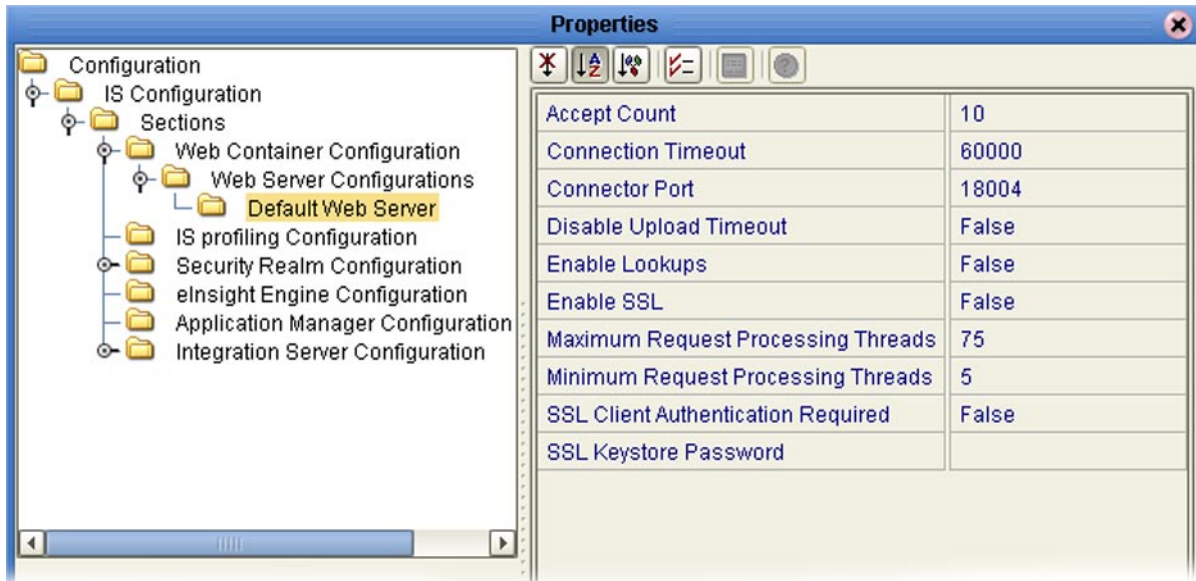
To create a new set of Web server configuration properties

- 1 Right-click **Web Server Configurations** to display the context menu, and click **Create New Section**.
- 2 Give the newly created section an appropriate name.
- 3 In the properties for the newly created section, make any necessary changes to the default properties. The **port** number must be different for each Web Server.
- 4 Click **OK**.

## Default Web Server

The Default Web Server properties serve as a template for the individual Web servers.

**Figure 109** Default Web Server Properties



**Table 42** Default Web Server Properties List

Property	Description
Accept Count	Specifies the maximum acceptable number of incoming connection requests when all possible request processing threads are in use. Any requests received beyond this number when the queue is full are refused. The default value is <b>10</b> .
Connection Timeout	Specifies the time period in milliseconds that this connector will wait for the request URI line to be presented, after accepting the connection. The default value is <b>60,000 ms</b> (1 min).
Connector Port	Specifies the connection port for the Web server. The port numbers are dynamically assigned; the number shown is typical.
Disable Upload Timeout	Allows the servlet container to use a different, and longer, connection timeout while a servlet is executing. This gives the servlet a longer time to complete execution, and/or provides a longer timeout during data upload. The default value is <b>False</b> .
Enable Lookups	If set to <b>True</b> , calls are made requesting getRemoteHost() to perform DNS lookups in order to return the actual host name of the remote client. If set to <b>False</b> , the DNS lookup is bypassed and the IP address is returned in string form, thereby improving performance. The default value is <b>False</b> .

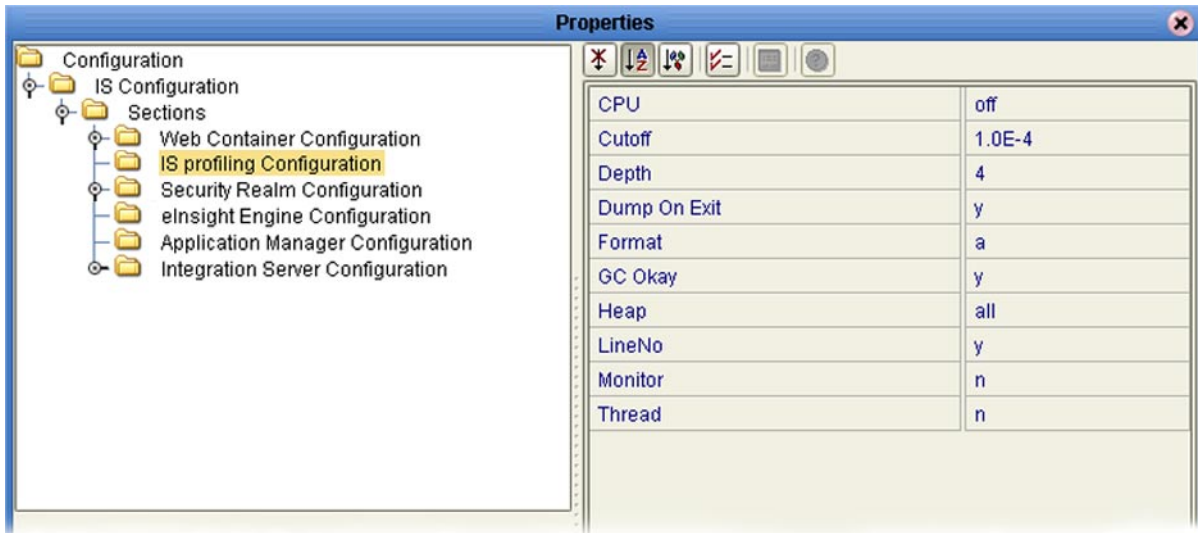
**Table 42** Default Web Server Properties List

Property	Description
Enable SSL	Specifies whether or not to enable the Secure Sockets Layer (SSL) protocol. The default value is <b>False</b> .
Maximum Request Processing Threads	Specifies the maximum number of request processing threads to be created by this connector, thereby determining the maximum number of simultaneous requests that can be handled. The default value is <b>75</b> .
Minimum Request Processing Threads	Specifies the number of request processing threads to be created by this connector when it is first started. This value must be less than the value set for the Maximum Request Processing Threads property. The default value is <b>5</b> .
SSL Client Authentication Required	Set this property to <b>True</b> if you want to require SSL Client Authentication. The default value is <b>False</b> .
SSL Keystore Password	Enter the password you used when creating the certificate (there is no default value).

## Performance Monitoring (Profiling) Configuration

You can monitor the performance of the integration server by using the built-in *Heap Analysis* tool, which is enabled and configured using the Profiling Configuration dialog box (see Figure 110).

**Figure 110** Profiling Configuration Properties



**Table 43** Profiling Configuration Properties List

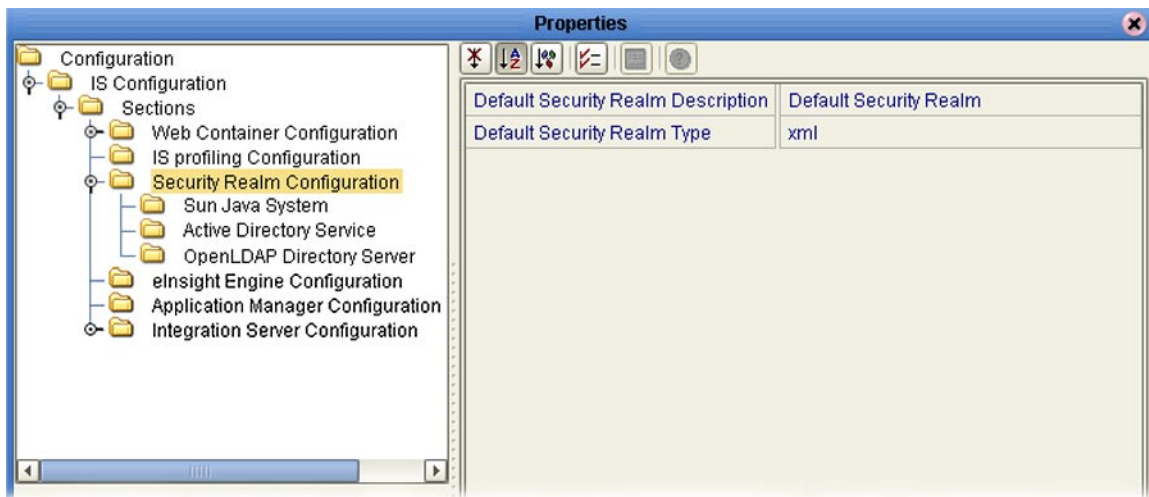
Property	Description
CPU	Specifies whether or not CPU usage is included in the trace. The default value is <b>off</b> .
Cutoff	Specifies the output cutoff point. The default value is <b>1.0E-4</b> .
Depth	Specifies the stack trace depth. The default value is <b>4</b> .
Dump on Exit	Specifies whether or not to dump on exit. The default value is <b>y</b> (yes).
Format	Specifies ASCII ( <b>a</b> ) or binary ( <b>b</b> ) output. The default value is <b>a</b> (ASCII).
GC Okay	Specified whether or not to allow garbage collection (GC) during sampling. The default value is <b>y</b> (yes).
Heap	Specifies the blocks of memory to include in traces. The default value is <b>all</b> .
LineNo	Specifies whether or not to include line numbers in traces. The default value is <b>y</b> (yes).
Monitor	Specifies whether or not to include monitor contention. The default value is <b>n</b> (no).
Thread	Specifies whether or not to include the thread in traces. The default value is <b>n</b> (no).



## Security Realm Configuration

These properties pertain to the Lightweight Directory Access Protocol (LDAP), if used. Subdirectories contain properties for SunONE Directory Server and Microsoft Active Directory Server. See the *eGate Integrator System Administration Guide* for information regarding Security Realm configuration.

**Figure 111** Security Configuration Properties



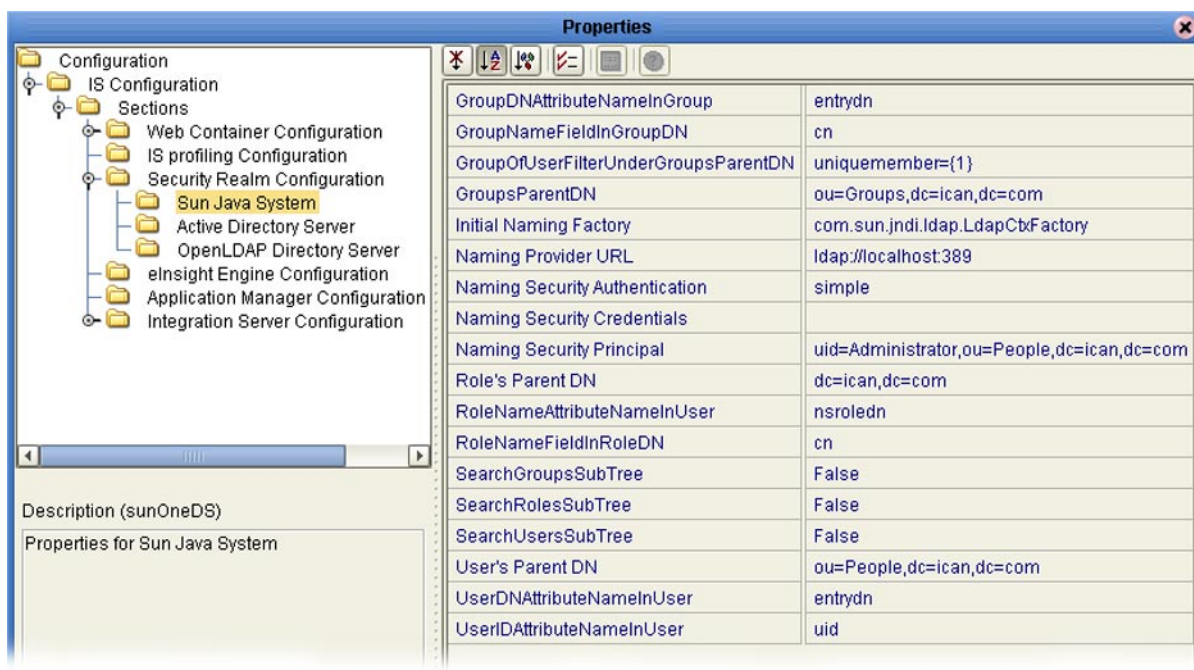
**Table 44** Security Realm Configuration Properties List

Property	Description
Default Security Realm Description	Specifies the name for the default LDAP Security Realm. The default value is <b>Default Security Realm</b> .
Default Security Realm Type	Specifies the default LDAP Security Realm type. The default value is <b>xml</b> .

## Security Realm Configuration: Sun Java System

These configuration properties are used only if you are using the Sun Java System (formerly SunONE) as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 112** Sun Java System Configuration Properties



**Table 45** Sun Java System Configuration Properties List

Property	Default Value
GroupDNAttributeNamelnGroup	entrydn
GroupNameFieldInGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	uniquemember={1}
GroupsParentDN	cn=Groups,dc=ican,dc=com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	simple
Naming Security Credentials	(password)
Naming Security Principle	uid=Administrator,ou=People,dc=ican,dc=com
Role's ParentDN	dc=ican,dc=com
RoleNameAttributeNamelnUser	nsroledn
RoleNameFieldInRoleDN	cn
SearchGroupsSubTree	False
SearchRolesSubTree	False

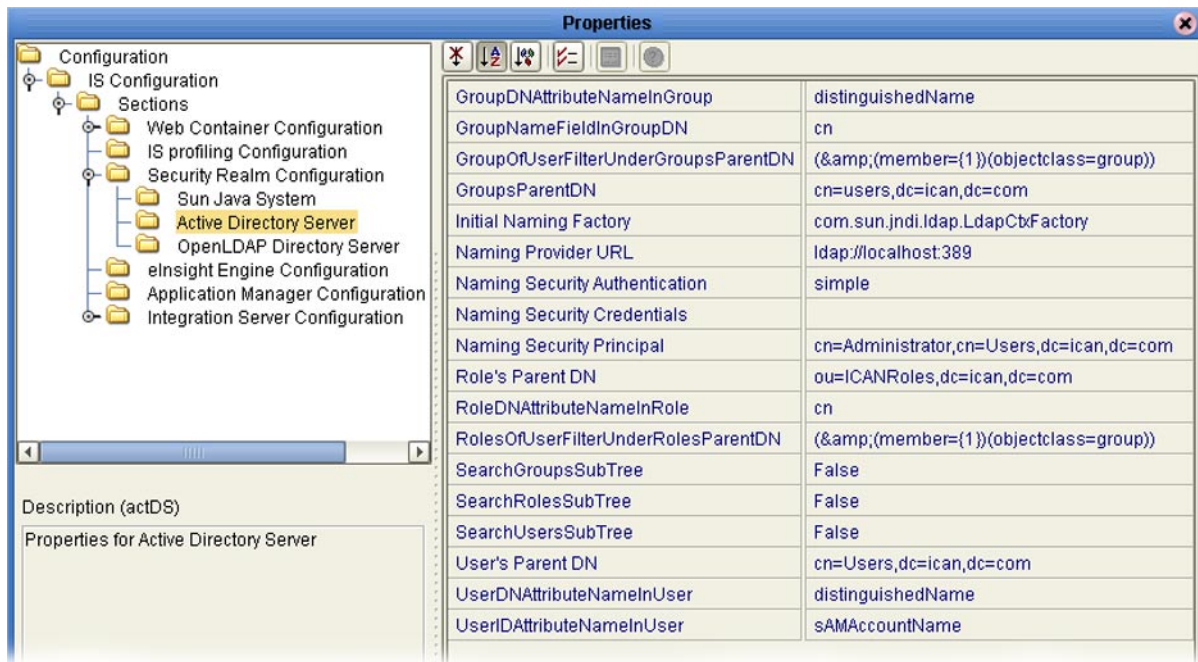
**Table 45** Sun Java System Configuration Properties List

<b>Property</b>	<b>Default Value</b>
SearchUsersSubTree	False
User's ParentDN	cn=People,dc=ican,dc=com
UserDNAttributeNameInUser	entrydn
UserIDAttributeNameInUser	uid

## Security Realm Configuration: Active Directory Service

These configuration properties are used only if you are using the Microsoft Active Directory Server as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 113** Active Directory Server Configuration Properties



**Table 46** Active Directory Server Configuration Default Properties List

Property	Default Value
GroupDNAttributeNameInGroup	distinguishedName
GroupNameFieldInGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	(&(member={1})(objectclass=group))
GroupsParentDN	cn=users,dc=ican,dc=com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	simple
Naming Security Credentials	(password)
Naming Security Principle	cn=Administrator,cn=Users,dc=ican,dc=com
Role's Parent DN	ou=ICANRoles,dc=ican,dc=com
RoleDNAttributeNameInRole	cn
RolesOfUserFilterUnderGroupsParentDN	(&(member={1})(objectclass=group))
SearchGroupsSubTree	False
SearchRolesSubTree	False

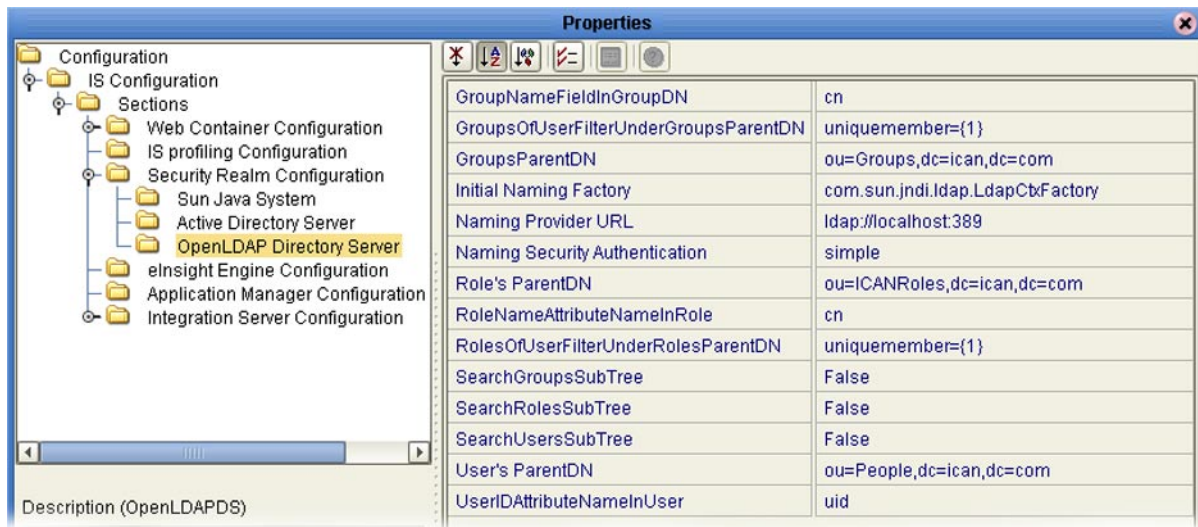
**Table 46** Active Directory Server Configuration Default Properties List

<b>Property</b>	<b>Default Value</b>
SearchUsersSubTree	False
User's Parent DN	cn=Users,dc=ican,dc=com
UserDNAttributeNameInUser	distinguishedName
UserIDAttributeNameInUser	sAMAccountName

## Security Realm Configuration: OpenLDAP Directory Server

These configuration properties are used only if you are using the OpenLDAP Directory Server as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 114** OpenLDAP Directory Server Configuration Properties



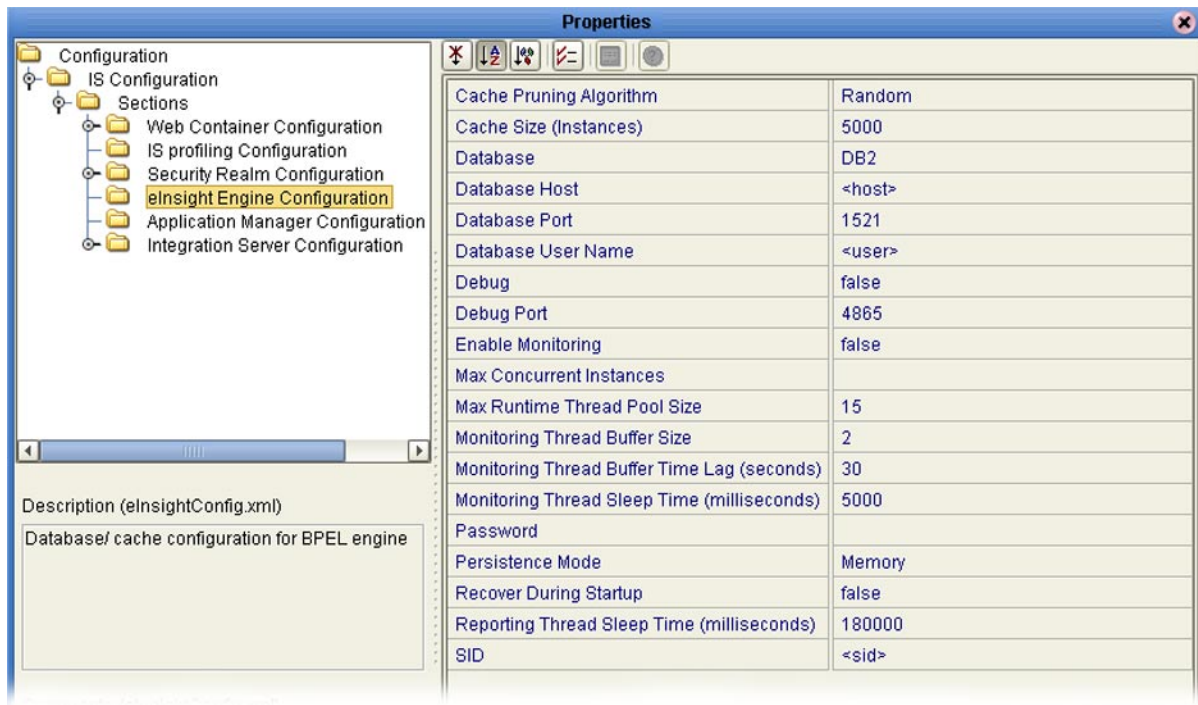
**Table 47** OpenLDAP Directory Server Configuration Default Properties List

Property	Default Value
GroupNameFieldInGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	uniquemember={1}
GroupsParentDN	ou=Groups,dc=ican,dc=stc.com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	simple
Role's ParentDN	ou=ICANRoles,dc=ican,dc=stc.com
RoleNameAttributeNameInRole	cn
RolesOfUserFilterUnderGroupsParentDN	uniquemember={1}
SearchGroupsSubTree	False
SearchRolesSubTree	False
SearchUsersSubTree	False
User's ParentDN	cn=People,dc=ican,dc=com
UserIDAttributeNameInUser	uid

## eInsight Engine Configuration

These configuration properties (see Figure 115) relate to the BPEL engine's database cache; see the *eInsight Enterprise Service Bus User's Guide* for information.

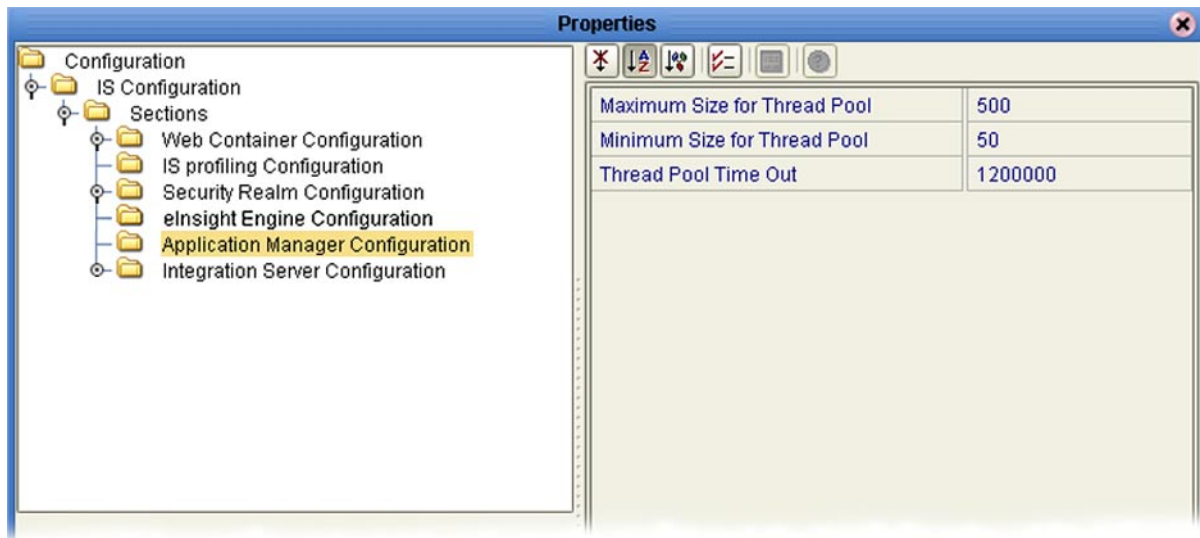
**Figure 115** eInsight Engine Configuration Properties



## Application Manager Configuration

You can set integration server thread pool variables using the Application Manager Configuration Properties dialog box (see Figure 116).

**Figure 116** Application Manager Configuration Properties



**Table 48** Application Manager Configuration Properties List

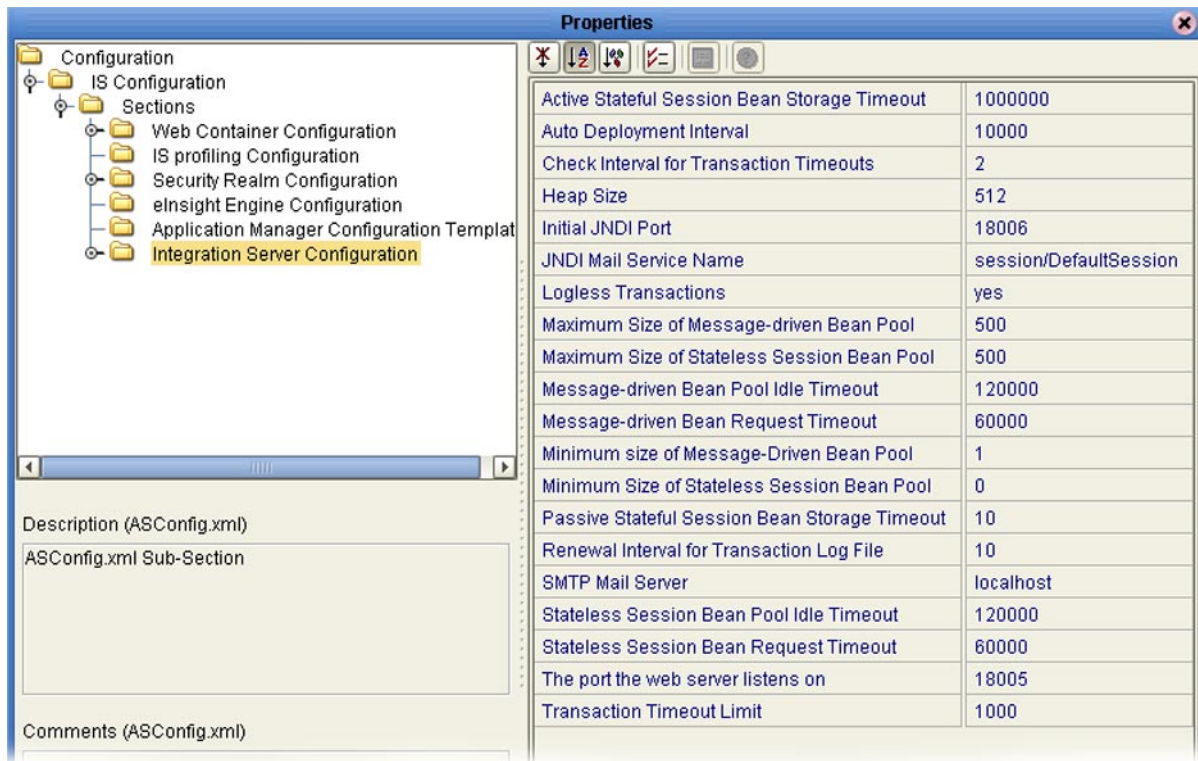
Property	Description
Maximum Size for Thread Pool	Specifies the maximum number of threads that can exist in the thread pool. The default value is <b>500</b> .
Minimum Size for Thread Pool	Specifies the minimum number of threads for the thread pool. The default value is <b>50</b> .
Thread Pool Time Out	Specifies the timeout interval for the thread pool, measured in milliseconds. The default value is <b>1,200,000</b> ms (20 min).



## Integration Server Configuration

Detailed, low-level configuration of the integration server is performed using the Integration Server Configuration Properties dialog box (see Figure 117).

**Figure 117** Integration Server Configuration Properties



**Table 49** Integration Server Configuration Properties List

Property	Description
Active Stateful Session Bean Storage Timeout	Specifies the interval after which an Active Stateful Session Bean is removed from storage, measured in minutes. The default value is <b>1,000,000</b> min, which ensures that it will not be removed unintentionally.
Auto Deployment Interval	Specifies the interval at which the auto-deployer checks the deployment directory for files, measured in milliseconds. The default value is <b>10,000</b> ms.
Check Interval for Transaction Timeouts	Specifies the interval between checks for transaction timeouts, measured in minutes. The default value is <b>2</b> min.
Heap Size	Specifies the Heap size in Megabytes; the minimum size is <b>512</b> Mb, which is the default value. Increasing this value increases the JVM size.
Initial JNDI Port	Specifies the initial port required by the Naming Service class for startup. The port numbers are dynamically assigned; the number shown is typical.

**Table 49** Integration Server Configuration Properties List

Property	Description
JNDI Mail Service Name	Specifies the name of the Java Naming and Directory Interface (JNDI) mail service. The default value is <b>session/DefaultSession</b> .
Logless Transactions	Specifies whether or not logless transactions are allowed. The default value is <b>yes</b> .
Maximum Size of Message-driven Bean Pool	Specifies the maximum number of Message-driven Beans allowed in the Message-driven Bean pool at one time. The default value is <b>500</b> .
Maximum Size of Stateless Session Bean Pool	Specifies the maximum number of Stateless Session Beans allowed in the Stateless Session Bean pool at one time. The default value is <b>500</b> .
Message-driven Bean Pool Idle Timeout	Specifies the timeout interval for the Message-driven Bean pool, measured in milliseconds. The default value is <b>120,000</b> ms (2 min).
Message-driven Bean Request Timeout	Specifies the interval after which a Message-driven Bean request times out, measured in milliseconds. The default value is <b>60,000</b> ms (1 min).
Minimum Size of Message-driven Bean Pool	Specifies the minimum number of Message-driven Beans allowed in the Message-driven Bean pool at one time. The default value is <b>1</b> .
Minimum Size of Stateless Session Bean Pool	Specifies the maximum number of Stateless Session Beans allowed in the Stateless Session Bean pool at one time. The default value is <b>1</b> .
Passive Stateful Session Bean Storage Timeout	Specifies the interval after which a Passive Stateful Session Bean is removed from storage, measured in minutes. The default value is <b>10</b> min.
Renewal Interval for Transaction Log File	Specifies the interval for renewing the Transaction Service log file, measured in hours. The default value is <b>10</b> hr.
SMTP Mail Server	Specifies the name of the SMTP mail host server. The default value is <b>localhost</b> . Note: Do not use underscore character in the host name.
Stateless Session Bean Pool Idle Timeout	Specifies the timeout interval for the Stateless Bean pool, measured in milliseconds. The default value is <b>120,000</b> ms (2 min).
Stateless Session Bean Pool Request Timeout	Specifies the interval after which a Stateless Bean request times out, measured in milliseconds. The default value is <b>60,000</b> ms (1 min).
The Port the Web Server Listens On	Specifies the port the Web server listens on. The port numbers are dynamically assigned; the number shown is typical.
Transaction Timeout Limit	Specifies the time limit for transactions to time out, measured in seconds. The default value is <b>1,000</b> sec.

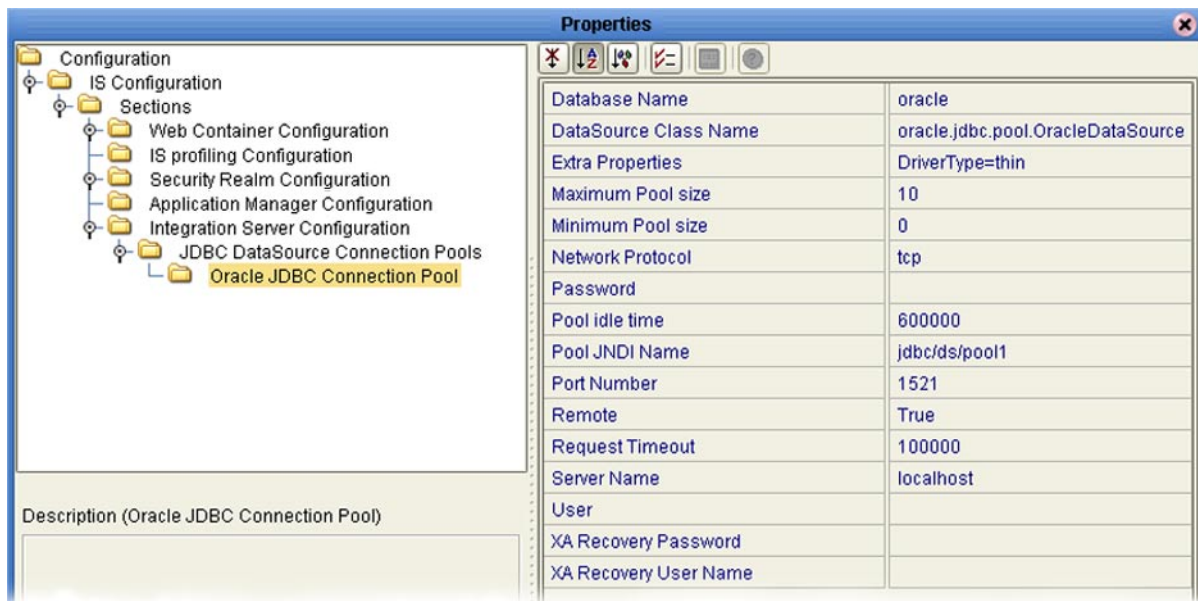
## JDBC DataSource Connection Pools

This directory serves as a container for the configuration properties for all JDBC DataSource Connection Pools associated with the selected integration server. The only set of default configuration properties currently furnished is for Oracle JDBC.

### Oracle JDBC Connection Pool

Connection Pool properties for an Oracle database associated with the integration server are specified in the Oracle JDBC Connection Pool dialog box (see Figure 118).

**Figure 118** Oracle JDBC Connection Pool Properties



**Table 50** Oracle JDBC Connection Pool Properties List

Property	Description
Database Name	Specifies the name of the database for which the pool is created. The default value is <b>oracle</b> .
DataSource Class Name	Specifies the name of the DataSource class. The default value is <b>oracle jdbc pool OracleDataSource</b> .
Extra Properties	Specifies custom properties for the DataSource, using semicolon-separated key-value pairs. The default value is <b>DriverType=thin</b> .
Maximum Pool Size	Specifies the maximum number of connections in the pool. The default value is <b>10</b> .
Minimum Pool Size	Specifies the minimum number of connections in the pool. The default value is <b>0</b> .
Network Protocol	Specifies the network protocol. The default value is <b>tcp</b> .

**Table 50** Oracle JDBC Connection Pool Properties List

Property	Description
Password	Specifies the password for the connection. There is no default value.
Pool Idle Time	Specifies the maximum time period in milliseconds that a connection may remain unused before it is removed from the pool in order to reduce the pool size. The default value is <b>600,000</b> ms (10 min).
Pool JINI Name	Specifies the unique Java Naming and Directory Interface (JNDI) name of the DataSource pool. The pool is bound in the java/namespac for local access or into the global namespace for remote access. The default value is <b>jdbc/ds/pool1</b> .
Port Number	Specifies the port number on which the server receives data. The default value is <b>1521</b> .
Remote	Specifies whether or not the DataSource should be bound into the global remote JINI namespace for access by remote clients. The default value is <b>true</b> .
Request Timeout	Specifies the maximum time period in milliseconds that a request for connection from the pool may block all other connections currently in use. The default value is <b>100,000</b> ms.
Server Name	Specifies the host name of the database server or IP address where the database server is running. The default value is <b>localhost</b> . Note: Do not use underscore character in the host name.
User	Specifies the user name authorized for creating connections. There is no default value.
XA Recovery Password	For XA DataSources only, specifies the password to use for XA transaction recovery. There is no default value.
XA Recovery User Name	For XA DataSources only, specifies the user name to use for XA transaction recovery. There is no default value.

## 7.5.2 Deploying User-Defined Stateless Session Beans

User-defined stateless session beans can be deployed to the SeeBeyond Integration Server following the procedure outlined in this section.

**Note:** *The deployment of stateful session beans, entity beans, and message-driven beans is not currently supported.*

### To deploy a stand-alone SLSB to the SeeBeyond Integration Server

- 1 Create and compile the EJB.
- 2 Write the `ejb-jar.xml` and `seebeyond-ejb.xml` deployment descriptors for your EJB.
- 3 Create a `.jar` file with the deployment descriptors in the `\META-INF` directory and the code in the root.
- 4 Move the `.jar` file into the `\logicalhost\stcis\deploy\new\integration_server_name` directory for deployment. The Integration Server will automatically pick up the `.jar` file from this location and deploy the EJB.

Examples of the EJBs and associated `.xml` files are as follows.

#### Example Remote Interface

```
package ejb.CustomApp;
import java.rmi.RemoteException;
import java.rmi.Remote;
import javax.ejb.*;

public interface CustomApp extends EJBObject, Remote
{
    public String getId() throws RemoteException;
}
```

#### Example Home Interface

```
package ejb.CustomApp;

import javax.ejb.*;
import java.rmi.Remote;
import java.rmi.RemoteException;
import java.util.*;

public interface CustomAppHome extends EJBHome
{
    public CustomApp create() throws CreateException, RemoteException;
}
```

#### Example Stateless Session Bean (SLSB)

```
package ejb.CustomApp;

import javax.ejb.*;
import java.io.Serializable;
import java.util.*;
import java.rmi.*;
import javax.naming.Context;
import javax.naming.InitialContext;

// import additional classes as needed "CustomController"
```

```

public class CustomAppBean implements SessionBean
{
    private SessionContext ctx;
    private CustomController mCustom;

    public void setSessionContext( SessionContext context )
    {
        this.ctx = context;
    }

    public void ejbCreate()
    {
        try {
            javax.naming.Context context = new InitialContext();
            // lookup Custom application
            Object ref = context.lookup("ejb/CustomController");
            CustomControllerHome CustomHome =
            (CustomControllerHome)javax.rmi.PortableRemoteObject.narrow(ref,
            CustomControllerHome.class);
            mCustom = CustomHome.create();
        } catch (Exception e) {
            System.out.println( e.getMessage() );
        }
    }

    public String getId()
    {
        SystemObjectPK key = new SystemObjectPK( "SBYN", "0000000001" );
        String EUID= "Not Found";
        try {
            EUID = mCustom.getEUID( key );
        }
        catch (Exception e) {
            System.out.println("===> Exception: " );
            System.out.println( e.getMessage() );
        }

        return( EUID );
    }

    // add additional EJB methods
}

```

### Example ejb-jar.xml file for the above SLSB

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE ejb-jar PUBLIC "-//Sun Microsystems, Inc.//DTD Enterprise
JavaBeans 2.0//EN" 'http://java.sun.com/dtd/ejb-jar_2_0.dtd'>
<!-- Generated XML! -->
<ejb-jar>
    <display-name>ServiceBeans</display-name>
    <enterprise-beans>
        <session>
            <description><![CDATA[Custom App Session Bean]]></
description>
            <display-name>Custom App</display-name>
            <ejb-name>CustomApp</ejb-name>
            <home>ejb.CustomApp.CustomAppHome</home>
            <remote>ejb.CustomApp.CustomApp</remote>
            <ejb-class>ejb.CustomApp.CustomAppBean</ejb-class>
            <session-type>Stateless</session-type>
            <transaction-type>Bean</transaction-type>
            <ejb-ref>
                <ejb-ref-name>ejb/CustomController</ejb-ref-name>

```

```
        <ejb-ref-type>Session</ejb-ref-type>
<home>com.stc.eindex.ejb.Custom.CustomControllerHome</home>
        <remote>com.stc.eindex.ejb.Custom.CustomController</
remote>
        <ejb-link>CustomController</ejb-link>
    </ejb-ref>
</session>
</enterprise-beans>
</ejb-jar>
```

### Example seebeyond-ejb.xml file for the above SLSB

```
<sbyn-ejb-deployment-descriptor>
  <enterprise-beans>
    <session>
      <ejb-name>CustomApp</ejb-name>
      <jndi-name>ejb/CustomApp</jndi-name>
      <security>
        <authorize>no</authorize>
        <authenticate>no</authenticate>
        <security-audit>no</security-audit>
      </security>
      <pool-min>1</pool-min>
    </session>
  </enterprise-beans>
</sbyn-ejb-deployment-descriptor>
```

### SLSB Deployment verification

Examine the log file `\logicalhost\logs\stc_is_integration_server_name.log`. You should find text such as **“CustomApp (EJB) was successfully deployed”** confirming deployment.

### To remove a stand-alone SLSB from the SeeBeyond Integration Server

- 1 Shut down the Logical Host containing the Integration Server where the SLSB is deployed.
- 2 Remove the `.jar` file created in the preceding deployment procedure from the `\logicalhost\stcis\repository\applications\integration_server_name\EJB` directory.
- 3 Restart the Logical Host.

## 7.6 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). eInsight Enterprise Service Bus 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.

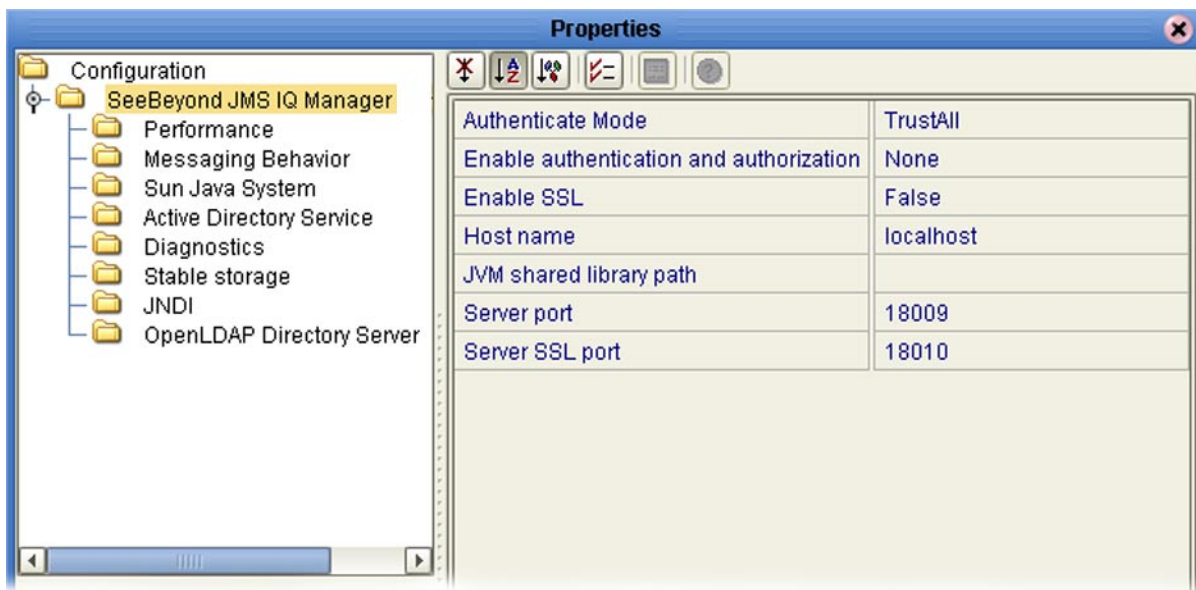
This section presents an overview of the configuration properties for the SeeBeyond JMS IQ Manager. For more information on the JMS IQ Manager, and deploying Project components to third-party message servers, see the *eGate Integrator JMS Reference Guide*.

### 7.6.1 SeeBeyond JMS IQ Manager Configuration

#### General Configuration

These properties cover the basic configuration of the JMS IQ Manager.

**Figure 119** JMS IQ Manager - General Configuration Properties





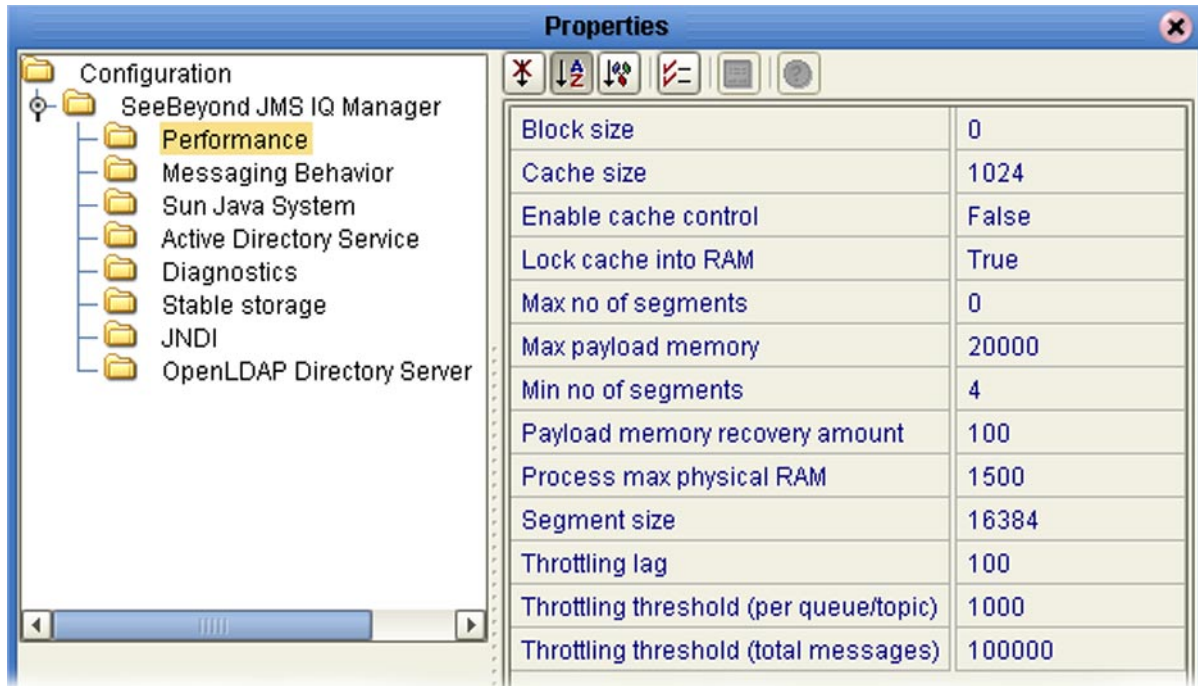
**Table 51** JMS IQ Manager - General Configuration Properties List

Property	Description
Authenticate Mode	Specifies whether or not the JMS IQ Manager authenticates components attempting to connect to it. The options are <b>Authenticate</b> and <b>TrustAll</b> ; the default value is <b>TrustAll</b> .
Enable authentication and authorization	Specifies whether or not security is enabled for the JMS IQ Manager, requiring a user name and password. The options are: <ul style="list-style-type: none"> <li>▪ <b>None</b> (no security— default value)</li> <li>▪ <b>File</b> (use File realm)</li> <li>▪ <b>SunONE</b> (use SunONE LDAP server)</li> <li>▪ <b>AD</b> (use Active Directory LDAP server)</li> </ul> If you specify either <b>SunONE</b> or <b>AD</b> , you must also configure your LDAP server. See <a href="#">Sun Java System</a> on page 165 and <a href="#">Active Directory Service</a> on page 167.
Enable SSL	Specifies whether or not to enable the Secure Sockets Layer (SSL) protocol for the JMS IQ Manager's TCP/IP connections. The options are <b>True</b> and <b>False</b> ; the default value is <b>False</b> .
Host name	Specifies the name of the host system for the JMS IQ Manager. The default value is <b>localhost</b> . Note: Do not use underscore character in the host name.
JVM shared library path	Specifies the path to the Java Virtual Machine shared library. The value depends on the operating system under which the Logical Host is running—use the ellipsis (...) button to select the correct value. The default value is <b>../jre/bin/client/jvm.dll</b> , for Windows platforms.
Server port	Specifies the TCP/IP port number. The port numbers are dynamically assigned; the number shown is typical. Each JMS IQ Manager must have a unique port number per system.
Server SSL port	Specifies the port on which the JMS IQ Manager listens for SSL connections. The port numbers are dynamically assigned; the number shown is typical.

## Performance

The Performance properties allow you to specify memory usage for optimum system performance. See the *eGate Integrator JMS Reference Guide* for additional information.

**Figure 120** Performance Configuration Properties



**Table 52** Performance Configuration Properties List

Property	Description
Block Size	Specifies the block size used for DBS files. Allowed values are <b>0</b> , <b>512</b> , and <b>1024</b> . If you specify <b>0</b> , the server will automatically determine the value to use by querying the operating system. <b>WARNING:</b> If you change this value, your existing DBS files will be unusable with the new setting.
Cache size	Specifies the total number of pages in the <i>read</i> cache (a page is 512 bytes on Windows and 1024 bytes on UNIX). The default value is <b>1024</b> pages (0.5 MB for Windows, 1 MB for UNIX).
Enable cache control	Specifies whether or not the JMS IQ Manager controls the cache synchronization to disk. The options are <b>True</b> and <b>False</b> ; the default is <b>False</b> .
Lock cache into RAM	Specifies whether or not the cache is locked into physical memory. The options are <b>True</b> and <b>False</b> ; the default is <b>True</b> .

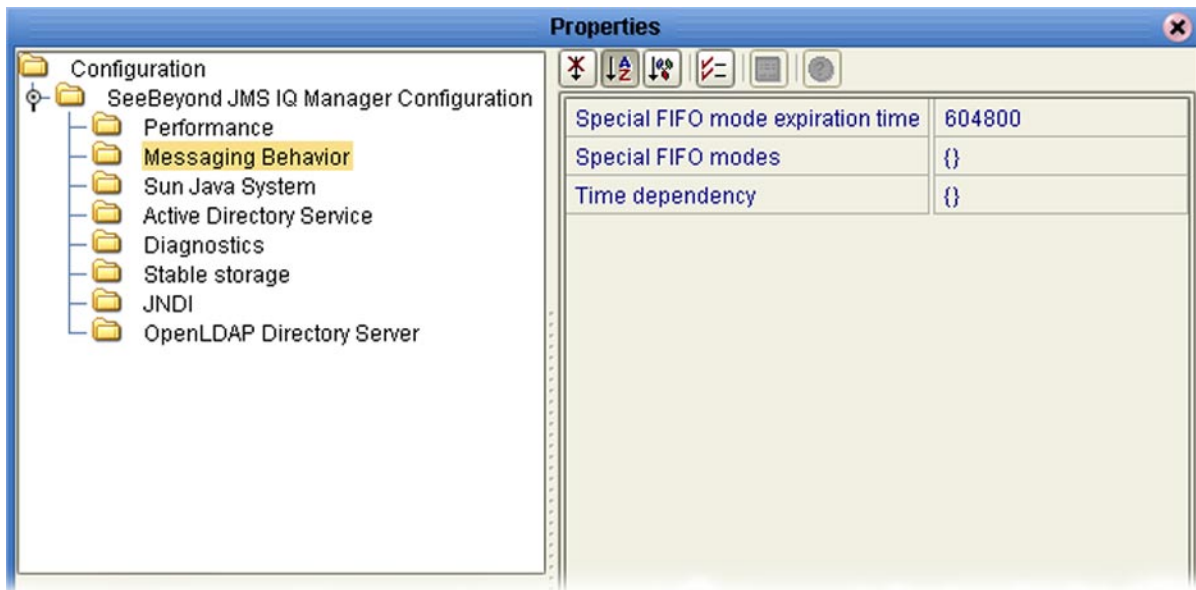
**Table 52** Performance Configuration Properties List

Property	Description
Max. no. of segments	Specifies the upper limit for the number of database files that the JMS IQ Manager creates for its stable message storage. The default value is <b>0</b> , which causes the JMS IQ Manager to create files as needed—limited only by disk space.
Max. payload memory	Specifies the upper limit for the amount of cache, in kB, allocated for the server to keep message payloads in cache. The default value is <b>20,000</b> kB.
Min. no. of segments	Specifies the minimum number of database files that the JMS IQ Manager initially creates for its stable message storage. The default value is <b>4</b> segments (files).
Payload memory recovery amount	Specifies the amount of cache, in kB, to recover in a recovery and cleanup operation. The default value is <b>100</b> kB.
Process max. physical RAM	Specifies the upper limit for the amount of RAM, in kB, allocated for use by the JMS IQ Manager as working memory (Windows platforms only). The default value is <b>1500</b> kB.
Segment size	Specifies the total number of pages in a single database file (segment). The default value is <b>16,384</b> pages (8 MB for Windows, 16 MB for UNIX).
Throttling lag	Specifies the number of messages that must be dequeued before message producers are no longer throttled. The default value is <b>100</b> .
Throttling threshold (per queue/topic)	Specifies the maximum number of messages after which all message producers for the message destination (topic or queue) are throttled. The default value is <b>1000</b> messages.
Throttling threshold (total messages)	Specifies the maximum number of messages for all message destinations, after which the JMS IQ Manager begins throttling producers. The default value is <b>100,000</b> messages.

## Messaging Behavior

The Messaging Behavior properties allow you to configure the order of message delivery. See the *eGate Integrator JMS Reference Guide* for additional information.

**Figure 121** Messaging Behavior Configuration Properties



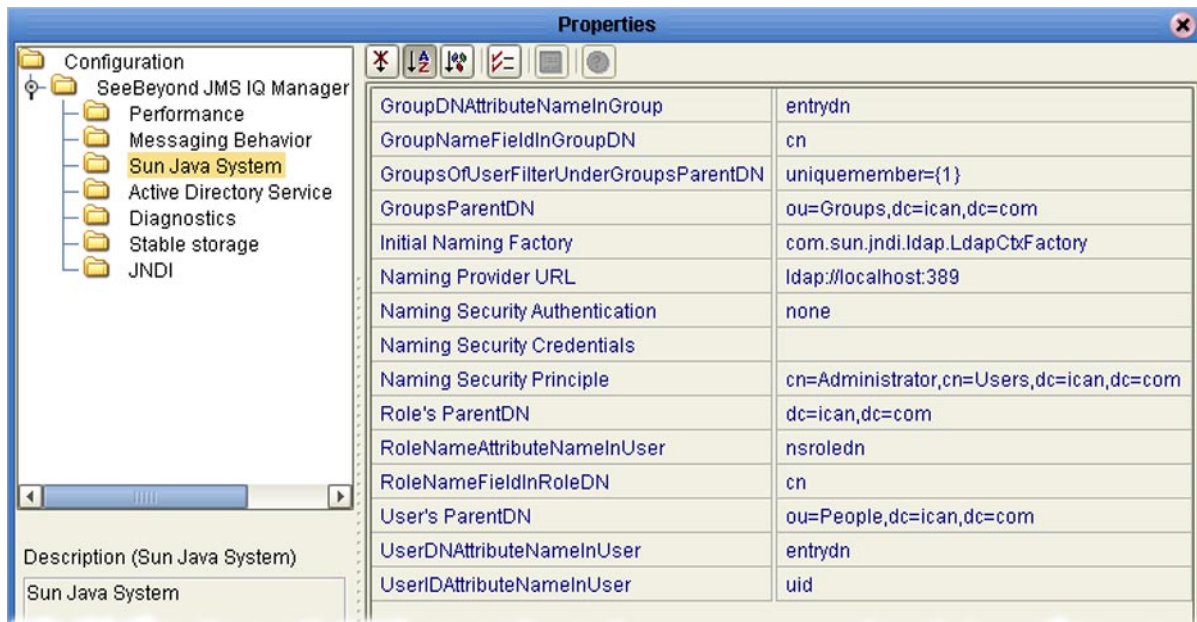
**Table 53** Messaging Behavior Configuration Properties List

Property	Description
Special FIFO mode expiration time	Specifies the expiration time in seconds for first-in, first-out (FIFO) mode operation. The default value is <b>604,800</b> seconds (7 days).
Special FIFO modes	Specifies the FIFO delivery order: <ul style="list-style-type: none"> <li>▪ fully concurrent (<b>0</b>)</li> <li>▪ protected concurrent (<b>1</b>)</li> <li>▪ fully serialized (<b>2</b>)</li> </ul> The default value is ( <b>0</b> ). See the <i>eGate Integrator JMS Reference Guide</i> for specification syntax.
Time dependency	Specifies whether or not the processing order is dependent on messages are associated with other destinations. The default value is <b>0</b> , indicating no such dependency. See the <i>eGate Integrator JMS Reference Guide</i> for syntax.

## Sun Java System

These configuration properties are used only if you are using the Sun Java System (formerly SunONE) as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 122** Sun Java System Configuration Properties



**Table 54** Sun Java System Configuration Default Properties List

Property	Default Value
GroupDNAttributeNamelnGroup	entrydn
GroupNameFieldlnGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	uniquemember={1}
GroupsParentDN	cn=Groups,dc=ican,dc=com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	none
Naming Security Credentials	(password)
Naming Security Principle	cn=Administrator,cn=Users,dc=ican,dc=com
Role's ParentDN	dc=ican,dc=com
RoleDNAttributeNamelnRole	nsroledn
RolesOfUserFilterUnderGroupsParentDN	cn
SearchRolesSubTree	False
SearchUsersSubTree	False

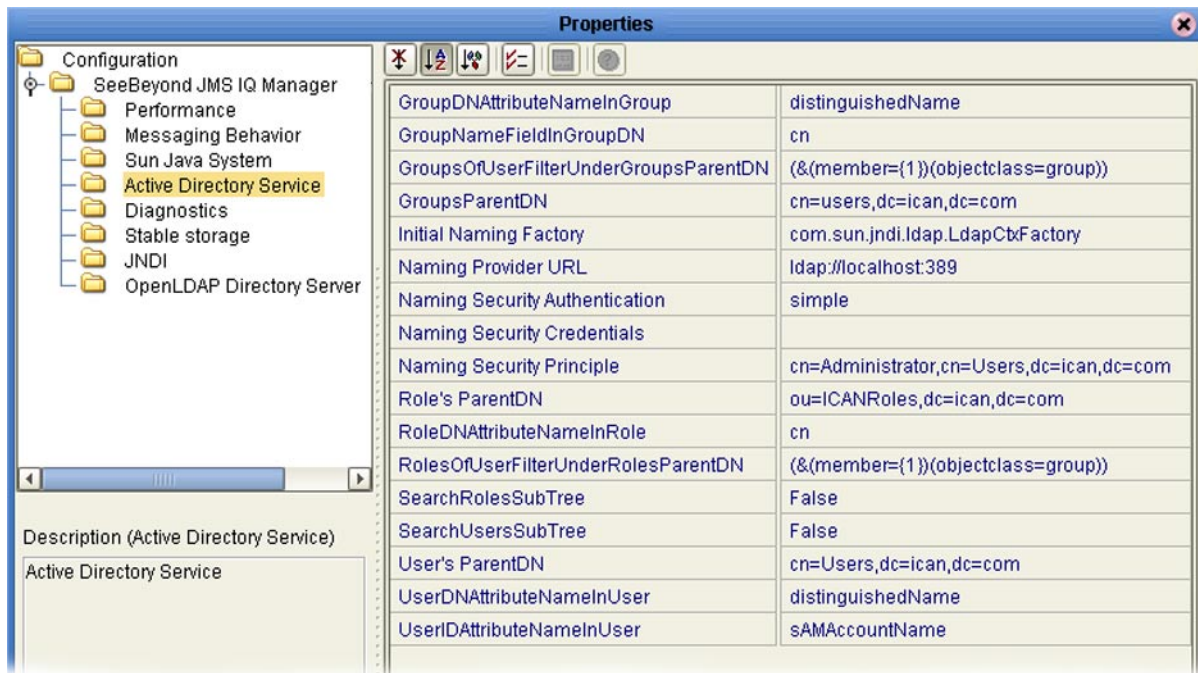
**Table 54** Sun Java System Configuration Default Properties List

Property	Default Value
User's ParentDN	cn=People,dc=ican,dc=com
UserDNAttributeNameInUser	entrydn
UserIDAttributeNameInUser	uid

## Active Directory Service

These configuration properties are used only if you are using the Microsoft Active Directory Service as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 123** Active Directory Service Configuration Properties



**Table 55** Active Directory Service Configuration Default Properties List

Property	Default Value
GroupDNAttributeNamelnGroup	distinguishedName
GroupNameFieldlnGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	(&(member={1})(objectclass=group))
GroupsParentDN	cn=users,dc=ican,dc=com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	simple
Naming Security Credentials	(password)
Naming Security Principle	cn=Administrator,cn=Users,dc=ican,dc=com
Role's ParentDN	ou=ICANRoles,dc=ican,dc=com
RoleDNAttributeNamelnRole	cn
RolesOfUserFilterUnderGroupsParentDN	(&(member={1})(objectclass=group))
SearchRolesSubTree	False

**Table 55** Active Directory Service Configuration Default Properties List

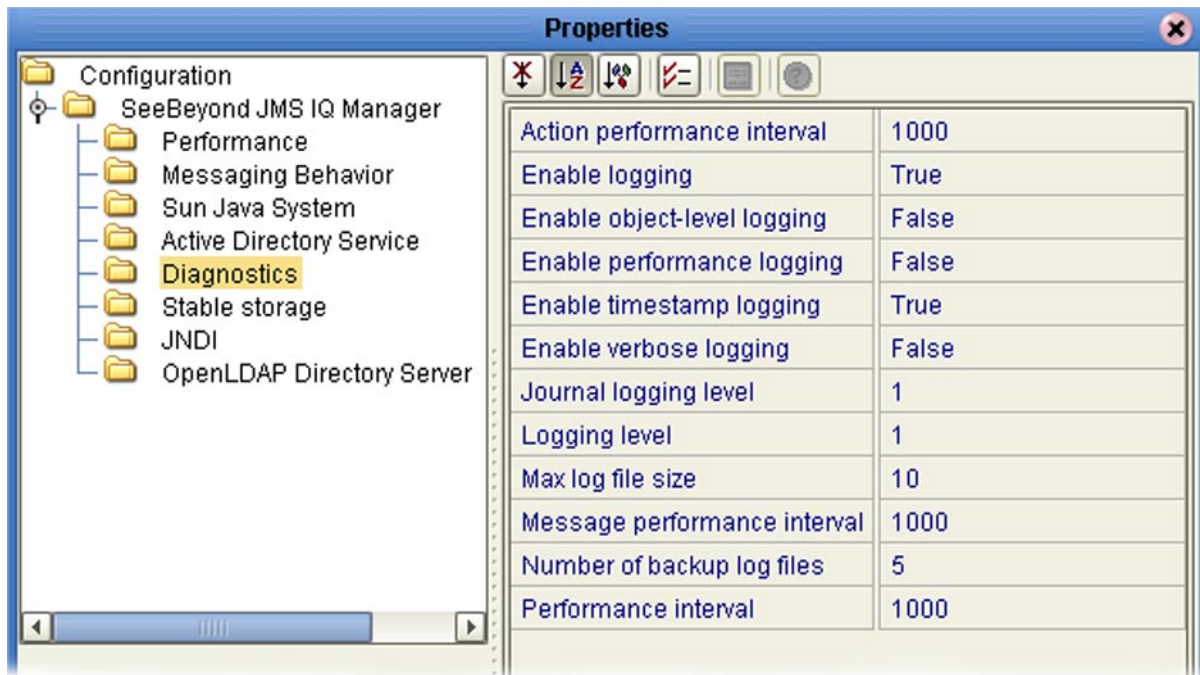
<b>Property</b>	<b>Default Value</b>
SearchUsersSubTree	False
User's ParentDN	cn=Users,dc=ican,dc=com
UserDNAttributeNameInUser	distinguishedName
UserIDAttributeNameInUser	sAMAccountName



## Diagnostics

The Diagnostics properties allow you to configure the logging operations for the JMS IQ Manager. See the *eGate Integrator JMS Reference Guide* for additional information.

**Figure 124** Diagnostics Configuration Properties



**Table 56** Diagnostics Configuration Properties List

Property	Description
Action performance interval	Specifies how many times the ActionManager's <i>Update</i> function is called before logging its performance statistics. The default is <b>1000</b> .
Enable logging	Specifies whether or not diagnostic information is written to the JMS IQ Manager log file. <b>True</b> (the default) enables logging, <b>False</b> disables it.
Enable object-level logging	Specifies whether or not object (constructor/deconstructor) information is written to the JMS IQ Manager log file. <b>True</b> enables logging, <b>False</b> (the default) disables it.
Enable performance logging	Specifies whether or not performance information is written to the JMS IQ Manager log file. <b>True</b> enables logging, <b>False</b> (the default) disables it.

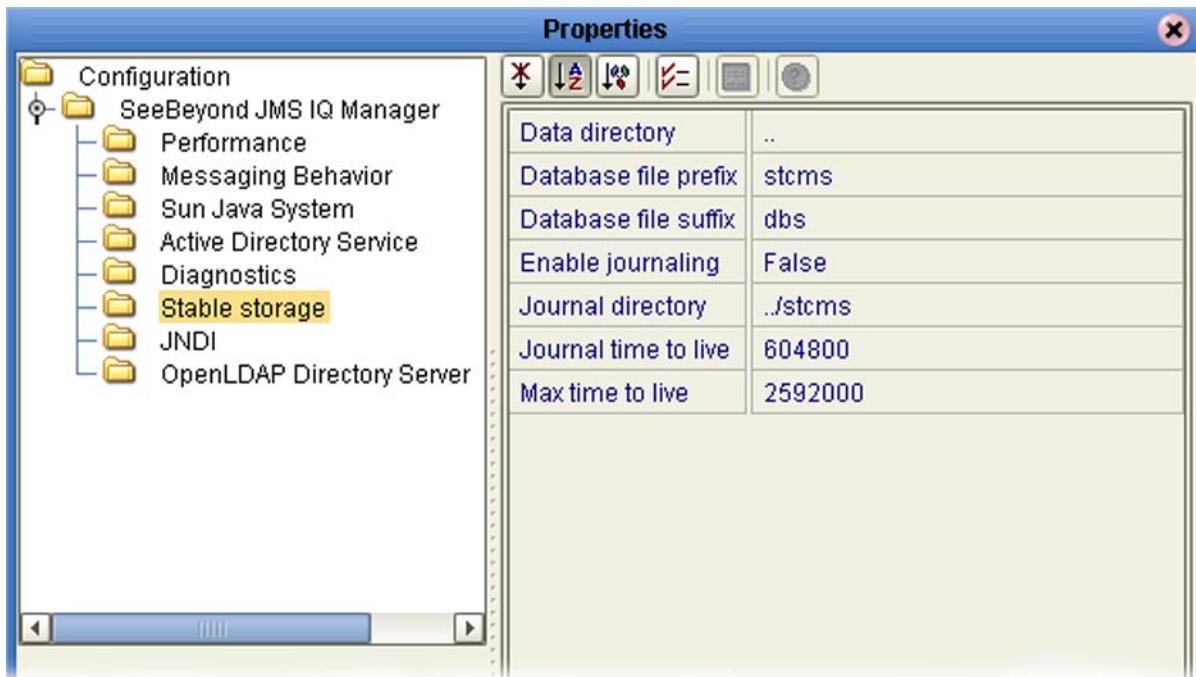
**Table 56** Diagnostics Configuration Properties List

Property	Description
Enable timestamp logging	Specifies whether or not timestamp information is written to the JMS IQ Manager log file. <b>True</b> (the default) enables logging, <b>False</b> disables it.
Enable verbose logging	Specifies whether or not full-length messages are written to the JMS IQ Manager log file. <b>True</b> enables logging, <b>False</b> (the default) disables it.
Journal logging level	Specifies the threshold severity level at which the system issues informational, warning, and error messages, and writes them to the JMS IQ Manager journal log. <ul style="list-style-type: none"> <li>▪ all messages (0)</li> <li>▪ warning, error, and fatal messages (1)</li> <li>▪ error and fatal messages (2)</li> <li>▪ fatal messages only (3)</li> </ul> The default level is (1).
Logging level	Specifies the threshold severity level at which the system issues informational, warning, and error messages, and writes them to the JMS IQ Manager log. <ul style="list-style-type: none"> <li>▪ all messages (0)</li> <li>▪ warning, error, and fatal messages (1)</li> <li>▪ error and fatal messages (2)</li> <li>▪ fatal messages only (3)</li> </ul> The default level is (1).
Max. log file size	Specifies the maximum size of the JMS IQ Manager log file, in MB. The default is <b>10</b> MB.
Message performance interval	Specifies the number of messages the JMS IQ Manager processes before logging its performance statistics. The default is <b>1000</b> .
Number of backup log files	Specifies the maximum number of JMS IQ Manager backup log files in the stack. The default is <b>5</b> .
Performance interval	Specifies the number of IMessages the IMessageManager creates before logging its performance statistics. The default is <b>1000</b> .

## Stable Storage

The Stable Storage properties allow you to configure the storage and journaling operations for the JMS IQ Manager. See the *eGate Integrator JMS Reference Guide* for additional information.

**Figure 125** Stable Storage Configuration Properties



**Table 57** Stable Storage Configuration Properties List

Property	Description
Data directory	Specifies the location for the JMS IQ Manager database files. The path can be either absolute or relative to the <b>logicalhost\</b> directory. The default is <b>..</b> (one directory up from the <b>logicalhost\</b> directory).
Database file prefix	Specifies the prefix for the JMS IQ Manager database file names. The default is <b>stcms</b> (for default file names of <i>stcms*.dbs</i> ).
Database file suffix	Specifies the extension for the JMS IQ Manager database file names. The default is <b>dbs</b> (for default file names of <i>stcms*.dbs</i> ).
Enable journaling	Enables or disables journaling, whereby every inbound message is automatically copied to the journal database. <b>True</b> enables, <b>False</b> disables; the default is <b>False</b> .

**Table 57** Stable Storage Configuration Properties List

Property	Description
Journal directory	Specifies the location for the journal database files. The path can be either absolute or relative to the <b>logicalhost\</b> directory. The default is <b>../stcms</b> .
Journal time to live	Specifies the maximum amount of time, in seconds, a journaled message persists before expiring. The default is <b>604,800</b> seconds (7 days).
Max. time to live	Specifies the maximum amount of time, in seconds, a live message persists before being removed from the queue. The default is <b>2,592,000</b> seconds (30 days).

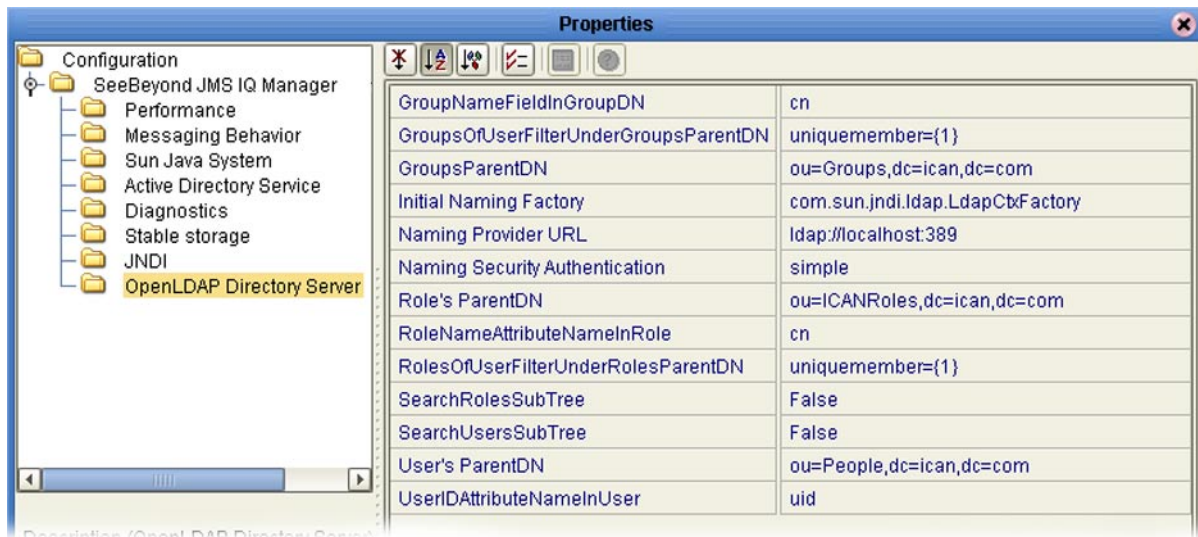
## JNDI

The JNDI configuration properties are not user-definable.

## OpenLDAP Directory Server

These configuration properties are used only if you are using the OpenLDAP Directory Server as an LDAP server. See the *eGate Integrator System Administration Guide* for additional information.

**Figure 126** OpenLDAP Directory Server Configuration Properties



**Table 58** OpenLDAP Directory Server Configuration Default Properties List

Property	Default Value
GroupNameFieldInGroupDN	cn
GroupOfUserFilterUnderGroupsParentDN	uniquemember={1}
GroupsParentDN	ou=Groups,dc=ican,dc=stc.com
Initial Naming Factory	com.sun.jndi.ldap.LdapCtxFactory
Naming Provider URL	ldap://localhost:389
Naming Security Authentication	simple
Role's ParentDN	ou=ICANRoles,dc=ican,dc=com
RoleNameAttributeNameInRole	cn
RolesOfUserFilterUnderGroupsParentDN	uniquemember={1}
SearchRolesSubTree	False
SearchUsersSubTree	False
User's ParentDN	ou=People,dc=ican,dc=com
UserIDAttributeNameInUser	uid

# Project Deployment

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

## What's in This Chapter

- [Deployment Profiles](#) on page 174
- [Deployment Editor](#) on page 175
- [Creating a Deployment Profile](#) on page 176
- [Automapping](#) on page 179
- [Mapping Variables](#) on page 180
- [Activating and Deactivating Deployment Profiles](#) on page 182
- [Deploying Projects to Third-Party Servers](#) on page 185

---

## 8.1 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:

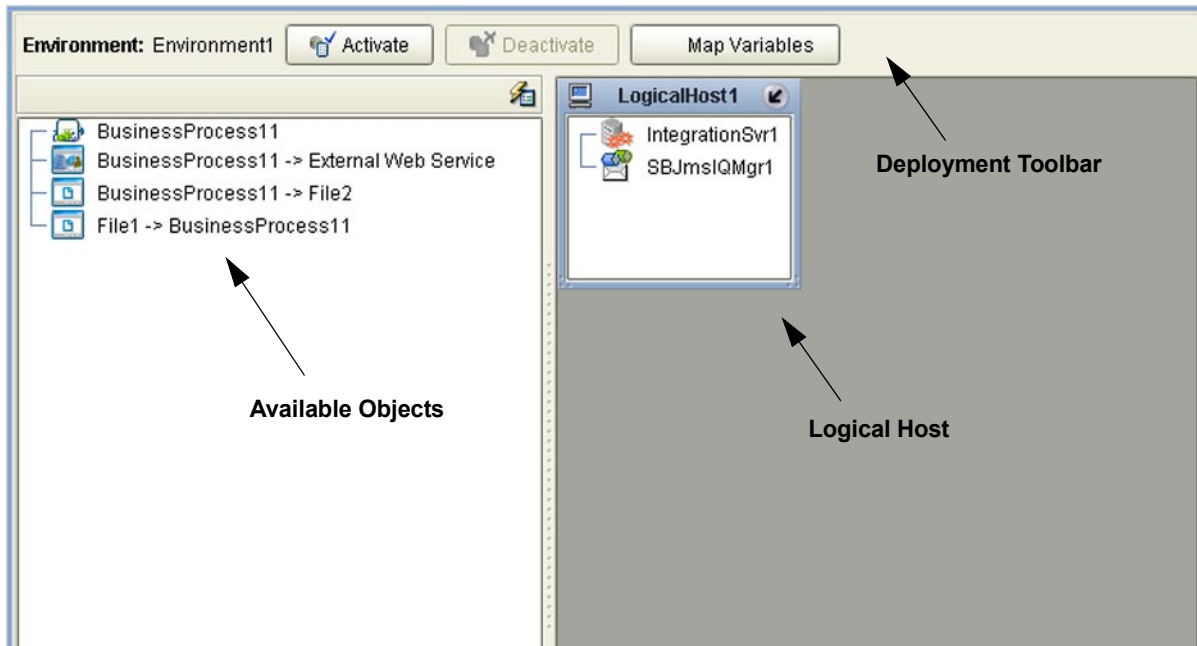
- Each Project can have zero or more Deployment Profiles, but each of a Project's active Deployment Profiles must be in a separate Environment.
- Each Environment can have zero or more Deployment Profiles assigned to it, but any given Environment can have only one Deployment Profile from a given Project.

Project deployment is illustrated in [Figure 2 on page 24](#).






## 8.2 Deployment Editor

The Deployment Editor (see Figure 127) allows you to create a new Deployment Profile or edit an existing one. To create a new Deployment Profile, right-click on a Project in the Project Explorer to display its context menu. From the menu, select **New > Deployment Profile**. To edit an existing Deployment Profile, simply click on its icon.

**Figure 127** Deployment Editor Window



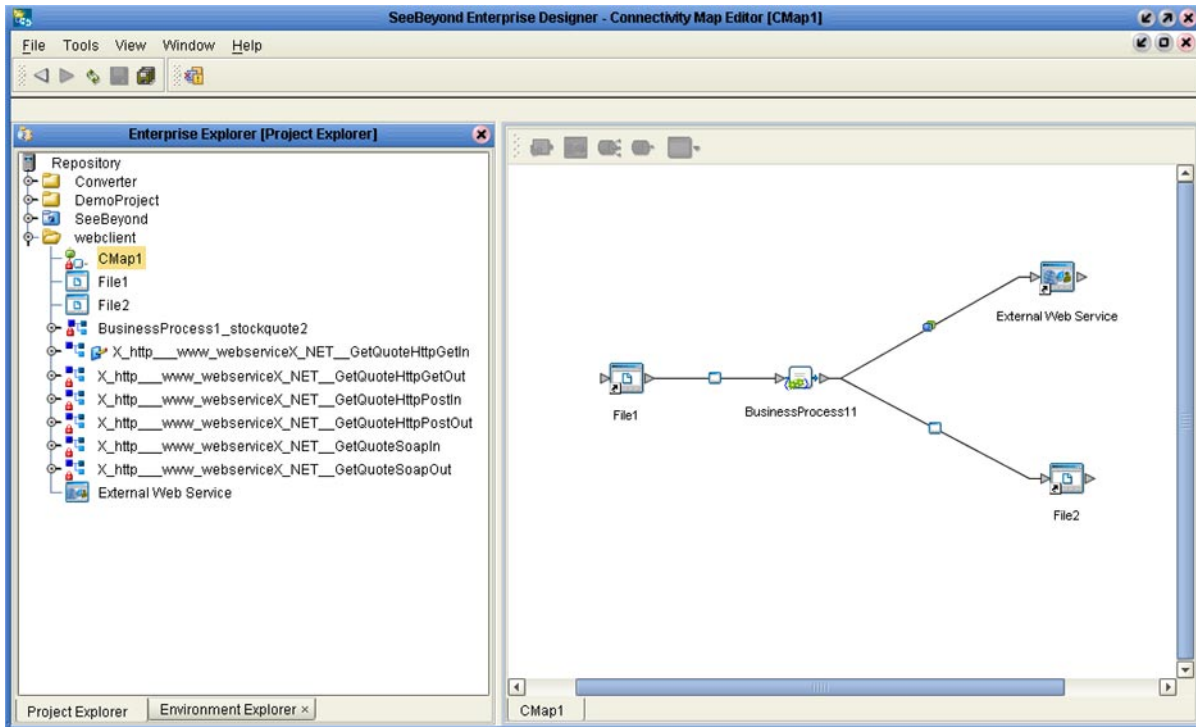
**Table 59** Deployment Toolbar Buttons

Button	Function
 Activate  Reactivate	Starts the Project by creating an enterprise archive (EAR) file based on the Connectivity Map and linking this file with the application server. After activation, button changes to <b>Reactivate</b> . See <a href="#">Activating and Deactivating Deployment Profiles</a> on page 182.
 Deactivate	Stops the Project by terminating the link between the EAR file and the application server, sets the Deployment Profile to <i>inactive</i> , and saves to the Repository. See <a href="#">Activating and Deactivating Deployment Profiles</a> on page 182.
 Map Variables	Allows you to assign names and values to Project variables for the specific Deployment Profile. See <a href="#">Mapping Variables</a> on page 180.
 <b>Auto Map</b>	Automatically deploys components to their matching containers, when there is a one-to-one correspondence between them. See <a href="#">Automapping</a> on page 179.

## 8.3 Creating a Deployment Profile

The Web Client Project shown in Figure 128 will be used as a deployment example.

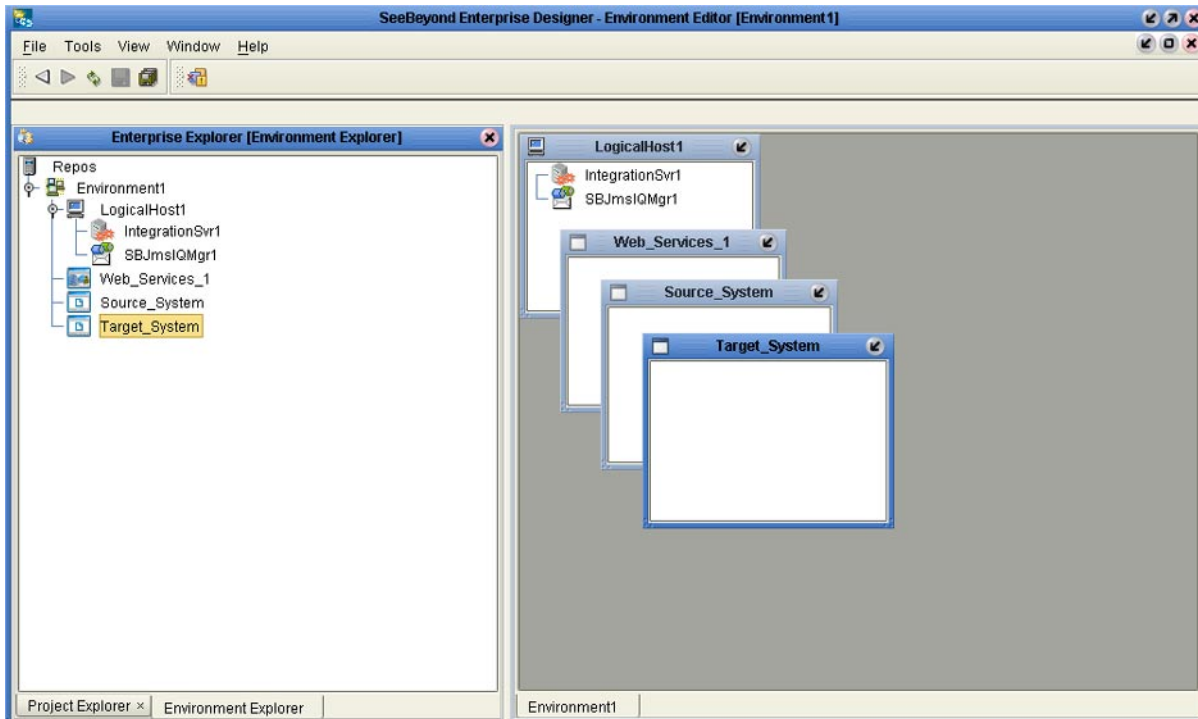
**Figure 128** Web Client Example Project



- 1 In the Environment Explorer, create an Environment and right-click on the Environment to display its context menu.
- 2 From the menu, select the Environment components you need and name them appropriately. They will appear as shown in Figure 129.

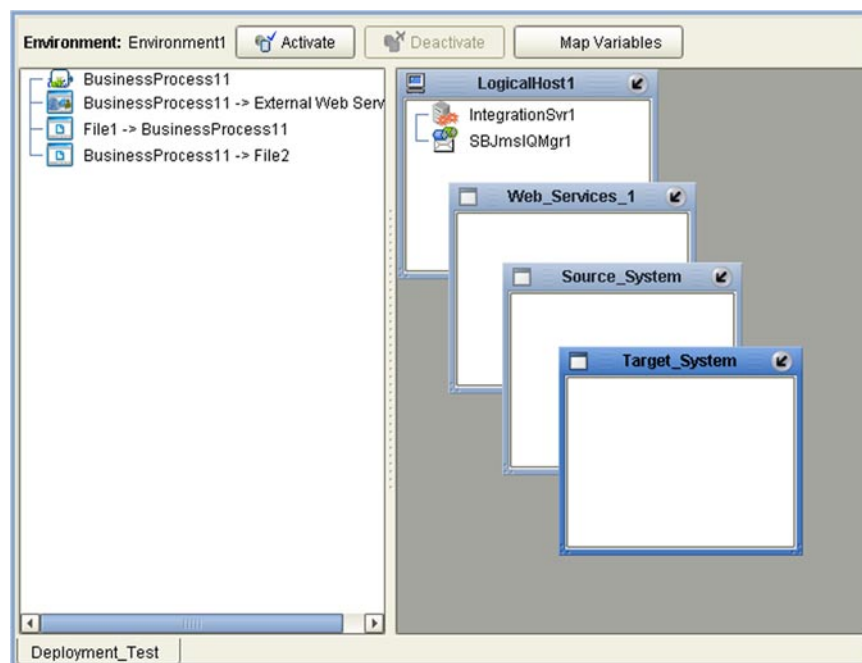


Figure 129 Web Client Example Environment



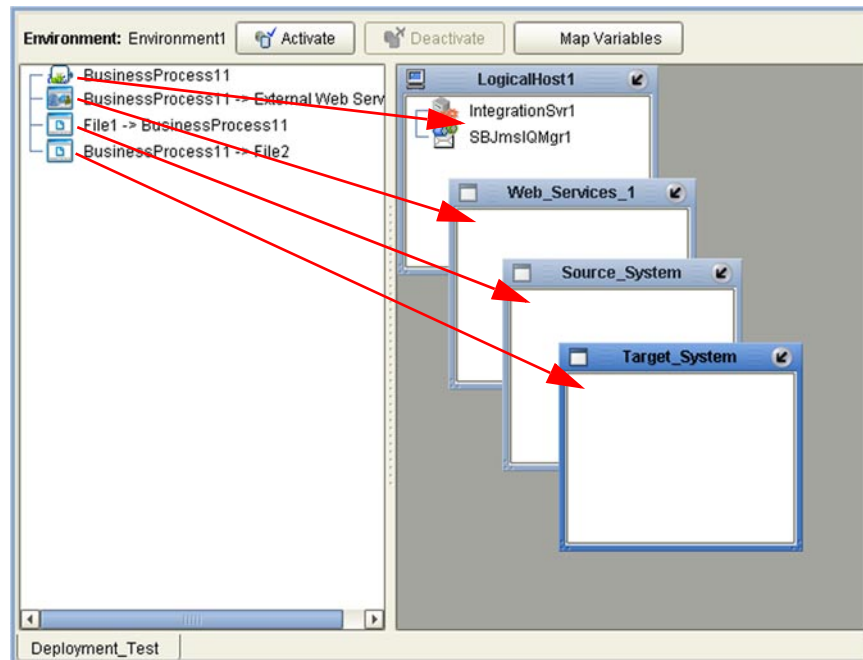
- 3 In the Project Explorer, right-click on the Project to display its context menu.
- 4 From the menu, select **New > Deployment Profile**. The Deployment Profile Editor appears, displaying the Environment you created (see Figure 130).

Figure 130 Example Deployment Profile (1)



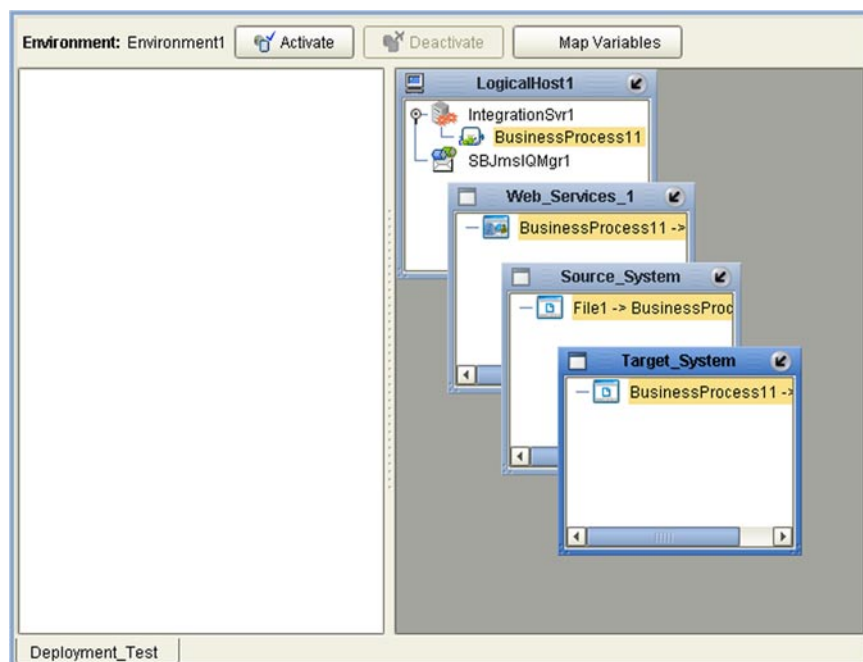
- 5 Drag the Project components from the left panel and drop them into the appropriate Environment components in the right panel, as illustrated in Figure 131.

**Figure 131** Example Deployment Profile (2)



- 6 When the Environment components are fully populated, the left panel will be blank, as shown in Figure 132. You should now **Save** the profile.

**Figure 132** Example Deployment Profile (3)



## 8.4 Automapping

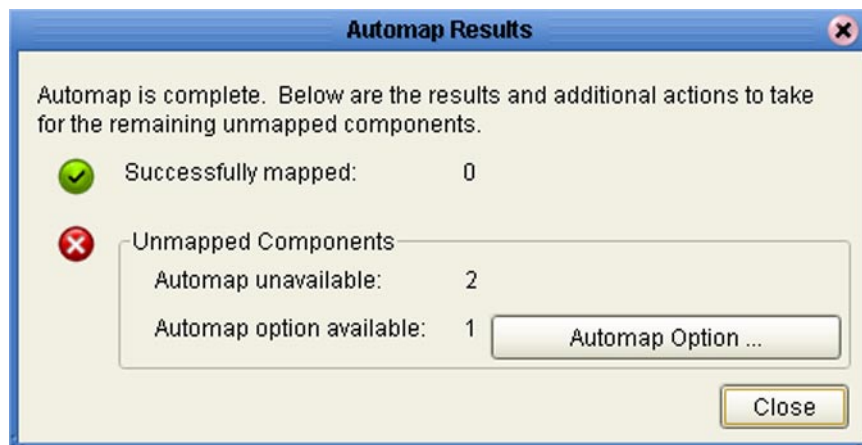
When a one-to-one correspondence exists between the available objects and the containers in the Environment to which you want to deploy them, clicking the **Auto Map** icon (see Figure 133) automatically deploys the components to their matching containers. (This feature only works with external systems for which it is enabled.)

**Figure 133** Auto Map Icon



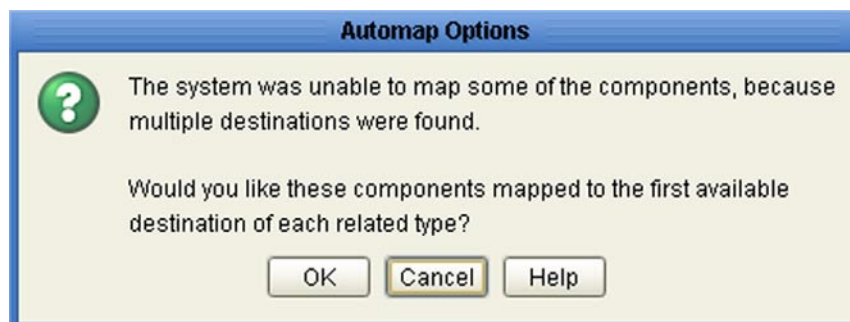
After the Automap feature executes, a dialog box is displayed showing the results (see Figure 134). The lack of an appropriate container, or ambiguity between potential containers, will cause a component to remain unmapped.

**Figure 134** Automap Results Dialog Box



In the case of an ambiguity, for example, mapping options are presented in an Automap Options dialog box (see Figure 135) that is displayed when you click **Automap Option**. You may click **OK** to accept the Automapping option, or click **Cancel** and map the deployable components manually.

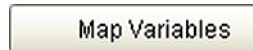
**Figure 135** Automap Options Dialog Box



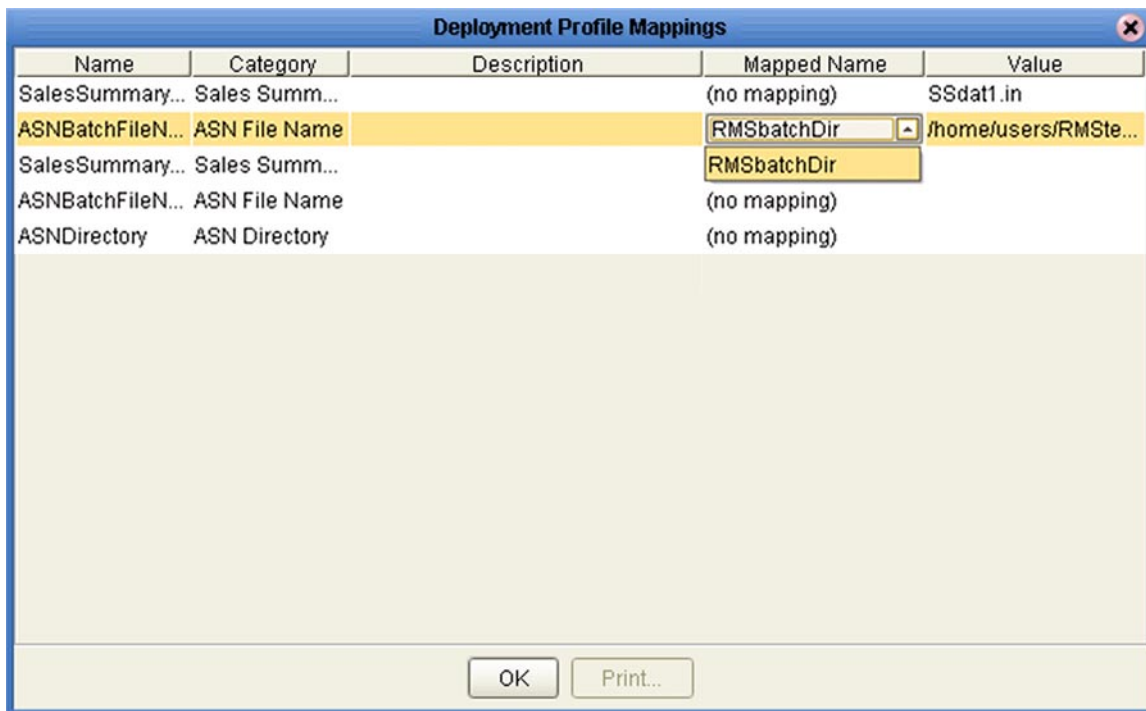
## 8.5 Mapping Variables

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 Variables** button displays the Deployment Profile Mappings panel, where you can assign names (see Figure 137) and values (see Figure 138).

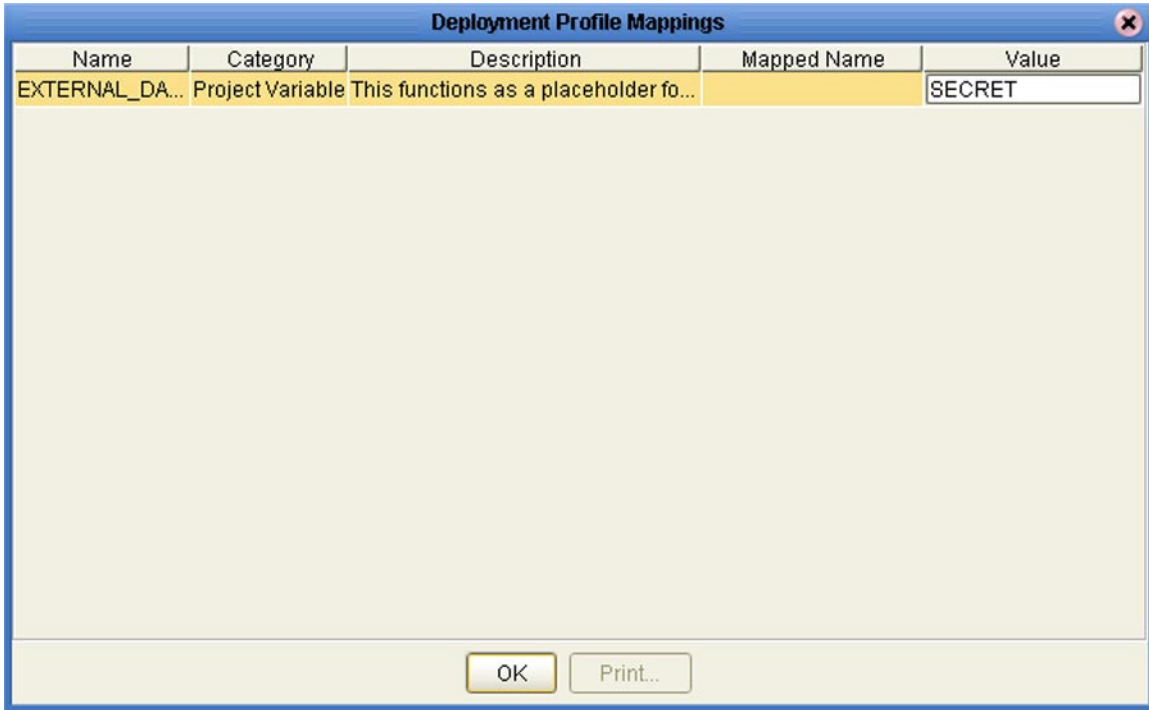
**Figure 136** Map Variables Button



**Figure 137** Deployment Profile Mappings



**Figure 138** Project Variable Value Entry



## 8.6 Activating and Deactivating Deployment Profiles

When activating or deactivating a Deployment Profile, you have the option of applying the changes to the Logical Host either immediately or at a later time. By activating the Deployment Profile without immediately applying the changes, you can check the validity of the entire Deployment Profile.

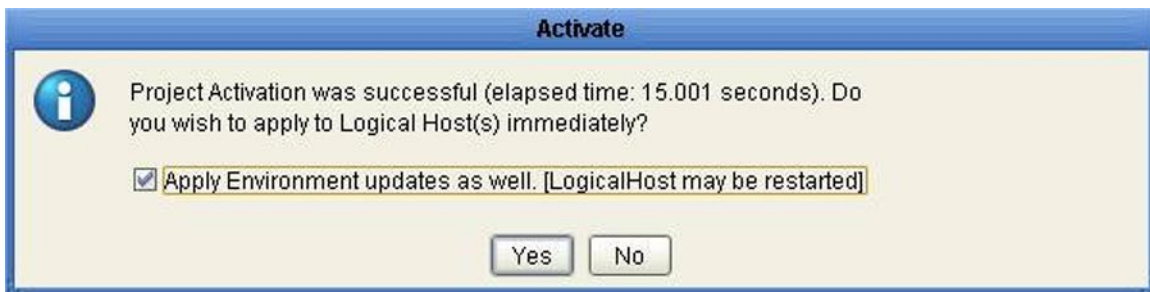
If you have multiple Deployment Profiles to deploy, you might work more efficiently by activating each of the Deployment Profiles without applying the changes, and then applying all of the changes to the Logical Host at a later time (*heed the warning given in the following procedures*).

### 8.6.1 Using Enterprise Designer

To activate a Deployment Profile

- 1 In the Project Explorer, select the Deployment Profile you want to activate.
- 2 Click the **Activate** button, which invokes the dialog box shown in Figure 139:

**Figure 139** Activate Dialog Box



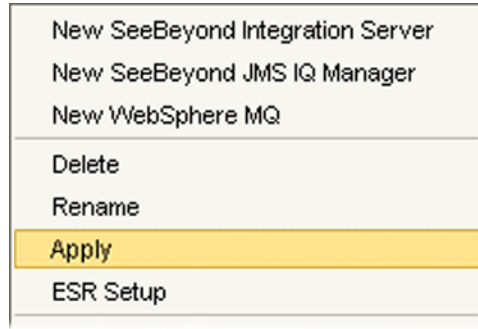
- 3 Select your response based on the following criteria:
  - If the Logical Host is running, and you wish to apply the changes immediately, click **Yes**.
  - If the Logical Host has not yet been bootstrapped, or you wish to apply the changes at a later time, click **No**.
- 4 If you click **Yes**, the information box shown in Figure 140 will appear after the changes have been applied. Click **OK** to proceed.

**Figure 140** Success Information Box



- 5 To apply the changes at a later time, right-click the Logical Host and select **Apply** from the menu (see Figure 141). This will download the latest configuration from the Repository to that Logical Host.

**Figure 141** Logical Host Context Menu - Apply



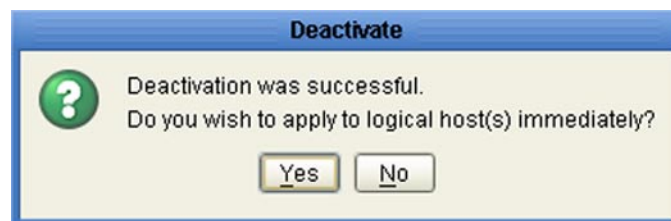
- 6 The information box shown in Figure 140 will appear after the changes have been completed. Click **OK** to proceed.

**Important:** Use the ICAN Monitor to verify that your changes have been applied before again running the apply command.

#### To deactivate a Deployment Profile

- 1 In the Project Explorer, select the Deployment Profile you want to deactivate.
- 2 Click the **Deactivate** button, which invokes the dialog box shown in Figure 142:

**Figure 142** Deactivate Dialog Box



- 3 Select your response based on the following 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**.
- 4 To apply the changes at a later time, right-click the Logical Host and select **Apply** from the menu (see Figure 141). This will apply all of the changes for that Logical Host.

**Important:** Use the ICAN Monitor to verify that your changes have been applied before again running the apply command.

## 8.6.2 Using a Command-line Script

A script named **CmdLineUtil.bat** (or **.sh**) allows you to deploy and undeploy projects via the command line. The *apply* and *unapply* commands provide the same function as clicking **Activate** or **Deactivate**, and then clicking **Yes**, in Enterprise Designer.

This command-line utility allows you three options for deployment:

- **Apply to all Logical Hosts in the Environment** (you must provide the Environment name).
- **Apply to a single Logical Host** (you must provide the Environment name and the Logical Host name).
- **Apply to a specific Deployment Profile** (you must provide the Project name and the Deployment Profile name).

The command-line utility also allows you to activate or deactivate from any computer, not only the one on which the Logical Host is installed. User name/password authentication is performed before the utility can be used.

This utility must be downloaded using Enterprise Manager as described in the *SeeBeyond ICAN Suite Installation Guide*. Before running the utility:

- The JAVA\_HOME environmental variable must be set on the host computer.
- The Repository must be running.
- You must first activate or deactivate the Project in Enterprise Designer.
- You must perform an initial bootstrap of the Logical Host(s) to download the components from the Repository.

**Important:** Use the ICAN Monitor to verify that your changes have been applied before again running the *apply* or *unapply* command.



## 8.7 Deploying Projects to Third-Party Servers

SeeBeyond's eInsight Enterprise Service Bus 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*.

Because of the versions of the Java Connection Architecture supported by WebLogic and WebSphere, the following restrictions apply:

- Services deployed to WebLogic or WebSphere are restricted to those internal to eInsight Enterprise Service Bus itself (between message destinations), and those associated with outbound eWays.
- Not all SeeBeyond eWays support third-party servers. Check the individual eWay User's Guides regarding such support, and also any additional configuration that may be necessary for compatibility with WebLogic or WebSphere.

### 8.7.1 BEA WebLogic

**Note:** Before using the WebLogic JMS, you must install additional *.jar* files as described below. For additional information, see the *eGate Integrator JMS Reference Guide*.

To install additional *.jar* files

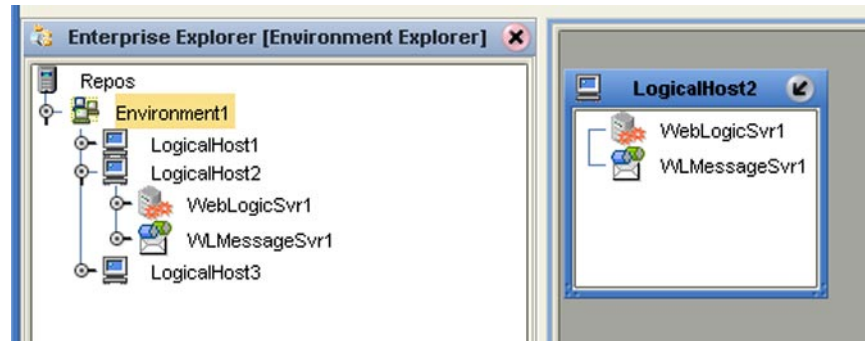
- 1 Download the **log4j.jar** file from the location below (this location may change).  
<http://jakarta.apache.org/log4j/docs>
- 2 Download the **xerces.jar** file from the location below (this location may change).  
<http://xml.apache.org/dist/xerces-j>
- 3 Place both *.jar* files into the `\weblogic8x\server\lib` directory.
- 4 Add the *.jar* files to the *set CLASSPATH* segment of the **startWLS.cmd** file located in the `\weblogic8x\server\bin` directory. The text to be added is:

```
%WL_HOME%\server\lib\log4j.jar;%WL_HOME%\server\lib\xerces.jar
```

To deploy an ICAN Project to a BEA WebLogic 8.0 or 8.1 environment

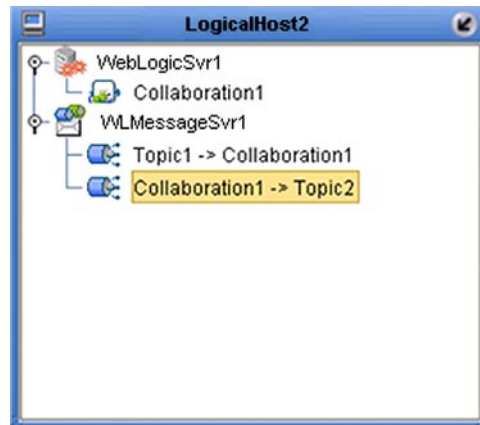
- 1 Create the following components in Enterprise Designer (see Figure 143):
  - A A new environment
  - B A Logical Host
  - C A WebLogic J2EE application server
  - D A WebLogic JMS message server

**Figure 143** WebLogic Deployment (1)



- 2 Create a new Deployment Profile to bind the Connectivity Map to the new WebLogic environment (see Figure 144).
  - A Drag the two topics and drop onto the WebLogic message server.
  - B Drag the Business Process and drop onto the WebLogic application server.

**Figure 144** 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\**WLEnvironmentName**\**ProjectName\_DeploymentProfileName.ear**

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

- 4 Start the BEA WebLogic server.
- 5 Navigate to **Server Administration Console > Deployments > Applications**.
- 6 Perform the following steps:

- A Add a new JMS Connection Factory.
- B Enter a JNDI name for the JMS Connection Factory:
 

```

jms\connectionfactory\xa-topic\  

LogicalHostName_MessageServerName

```

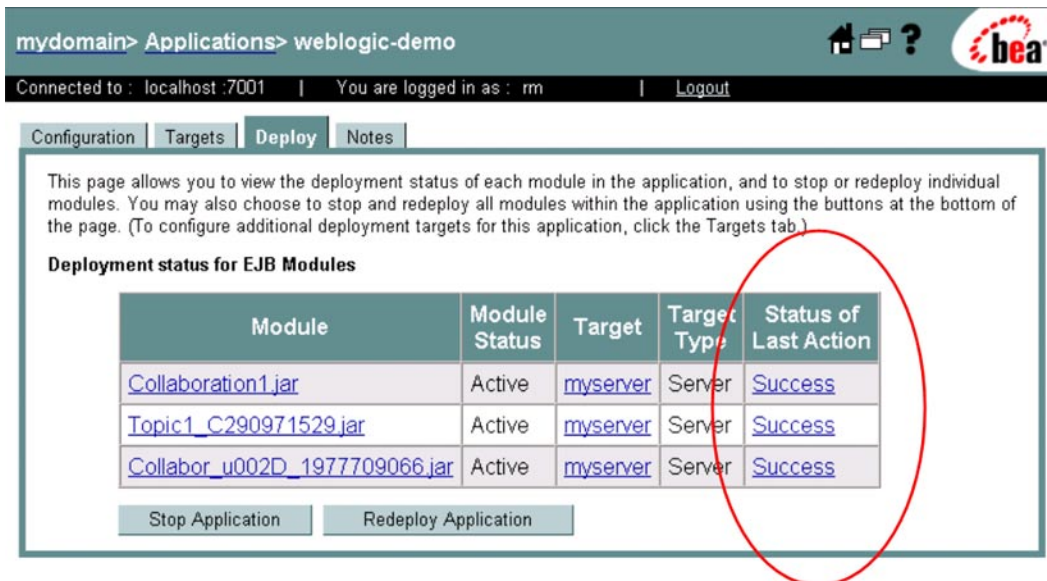
For example, the default name would be:

```

jms\connectionfactory\xa-topic\LogicalHost1_WLMessageSvr1

```
- C Verify that the WebLogic JMS Server Destination names for topics match those in eInsight Enterprise Service Bus.
- 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 145, which shows a WebLogic 8.1 example).

**Figure 145** WebLogic Deployment Verification



## 8.7.2 IBM WebSphere

**Note:** Before using the WebSphere JMS, you must install the **log4j.jar** file. For additional information, see the *eGate Integrator JMS Reference Guide*.

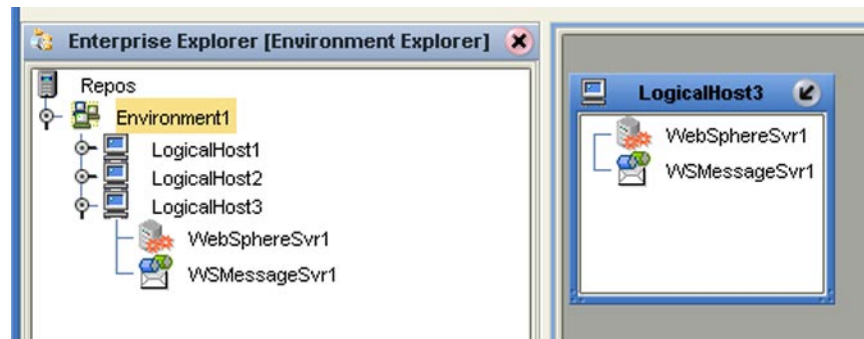
To install **log4j.jar**

- 1 Download the **log4j.jar** file from the location below (this location may change).  
<http://jakarta.apache.org/log4j/docs>
- 2 Place the **log4j.jar** file into the `\WebSphere\AppServer\lib` directory.

To deploy an ICAN Project to an IBM WebSphere 5.0.x or 5.1 environment

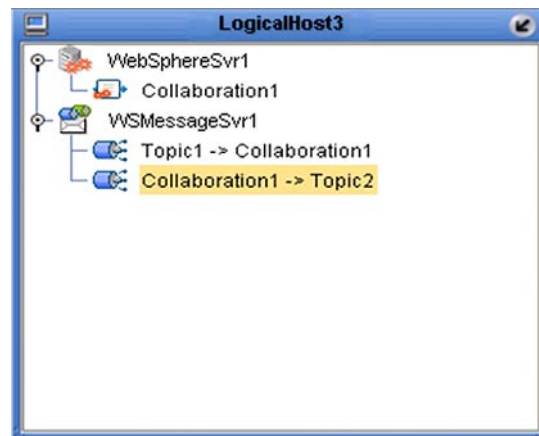
- 1 Create the following components in Enterprise Designer (see Figure 146):
  - A A new environment
  - B A Logical Host
  - C A WebSphere J2EE application server
  - D A WebSphere JMS message server

**Figure 146** WebSphere Deployment (1)



- 2 Create a new Deployment Profile to bind the Connectivity Map to the new WebSphere environment (see Figure 147).
  - A Drag the two topics and drop onto the WebSphere message server.
  - B Drag the Business Process and drop onto the WebSphere application server.

**Figure 147** 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_WSMMessageSvr1
```

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

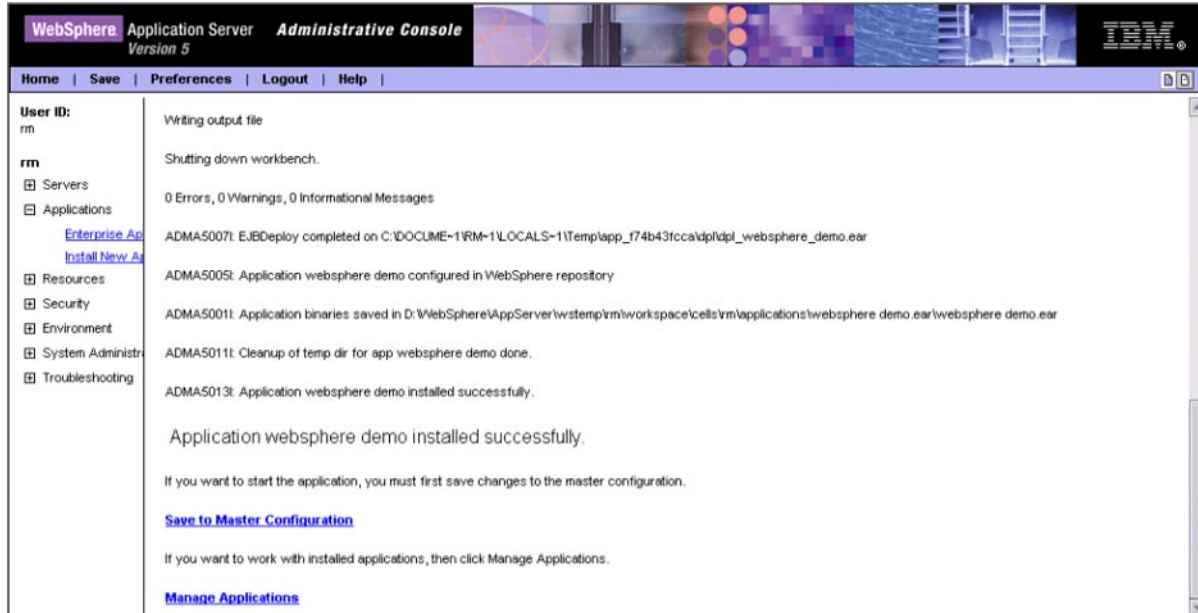
- 8 From the Administrative Console, navigate to **Applications > Enterprise Applications > Install New Application**.
- 9 In *Preparing for the application installation*:
  - A Enter the path for the EAR file described in step 3 and click **Next**.
  - B Select **Generate Default Bindings** and click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

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

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

- 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 148, which shows a WebSphere 5 example).

**Figure 148** WebSphere Deployment Verification



# Web Services

This chapter describes the use of the Web Services capability of eInsight Enterprise Service Bus.

## What's in This Chapter

- **Overview** on page 191
- **SeeBeyond Web Services** on page 192
- **UDDI Registry** on page 193
- **Web Services Application** on page 196
- **Web Services External System** on page 197
- **Load Balancing** on page 199
- **Building a Web Service** on page 202
- **Building a Web Services Client** on page 207
- **Setting Up Secure Web Services** on page 214

---

## 9.1 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, which are XML-based, 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.

---

## 9.2 SeeBeyond Web Services

eInsight Enterprise Service Bus provides the capability to create either a Web Service, which can be invoked by a remote client, or a Web Services Client to invoke a Web Service. A Business Process developed in eInsight Business Process Manager, for example, can be exposed as a Web Service. As such, it can invoke other external Web Services or be invoked by any Web Services Client. See [Building a Web Services Client](#) on page 207 and [Building a Web Service](#) on page 202.

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 112.
- **WSDL Editor**  
See the *eInsight Business Process Manager User's Guide*.
- **WSDL Interface Designer**  
See the *eInsight Business Process Manager User's Guide*.
- **WSDL Viewer**  
See the *eInsight Business Process Manager User's Guide*.
- **UDDI Registry**  
All ICAN objects represented in the Repository that can be accessed as Web services are presented via a UDDI-compliant server. See [UDDI Registry](#) on page 193.

ICAN Web Services adheres to guidelines and specifications developed by the Web Services-Interoperability Organization (WS-I) to the maximum possible extent. As a result, ICAN Web Services currently supports *document/literal* and *RPC/literal* messaging only, since the WS-I Basic Profile 1.1 rules out the use of *encoded* messaging.



## 9.3 UDDI Registry

In general, all ICAN objects that expose themselves as Web services, such as eInsight Business Processes, are listed in a UDDI registry. The deployment activation process automatically publishes entries to the UDDI registry and creates the necessary sections in the WSDL files to expose them as Web services.

This registry can be viewed on the SeeBeyond Web Services page (see Figure 149), part of Enterprise Manager. The URL of the SeeBeyond Web Services page is:

`http://hostname:portnumber/stcuddi`

**Note:** The *hostname* is the fully-qualified, network-addressable 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.

**Figure 149** SeeBeyond UDDI Registry

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

Each entry on the SeeBeyond Web Services page includes:

- The ICAN environment name.
- The actual (Web) Service name.
- The URL for the associated WSDL file.

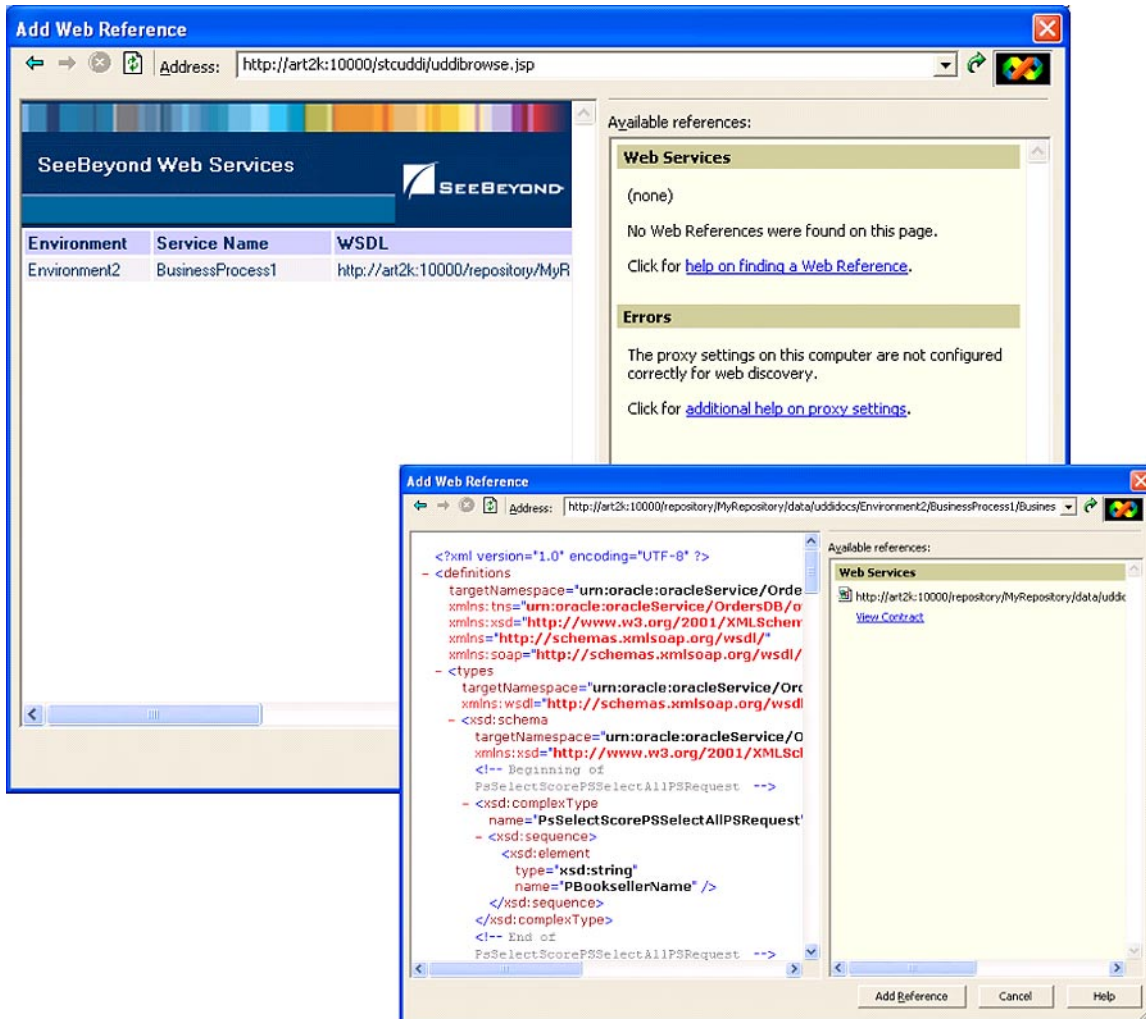
Select an entry (line item) to display its WSDL file, as shown in Figure 150.

**Figure 150** Example Web Service WSDL File

```
<?xml version="1.0" encoding="UTF-8" ?>
- <definitions targetNamespace="http://seebeyond/quoteservice" xmlns:tns="http://seebeyond/quoteservice"
  xmlns:sbynpk="http://bpel.seebeyond.com/hawaii/5.0/privateExtension/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
- <types targetNamespace="http://seebeyond/quoteservice">
  - <xsd:schema targetNamespace="http://seebeyond/quoteservice"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element type="xsd:string" name="QuoteRequest" />
    <xsd:element type="xsd:string" name="QuotePrice" />
    <xsd:element type="xsd:string" name="QuoteCompany" />
    </xsd:schema>
  </types>
- <message name="msgProvideQuoteRequest">
  <part name="TICKER" element="tns:QuoteRequest" />
</message>
- <message name="msgQuoteResponse">
  <part name="price" element="tns:QuotePrice" />
  <part name="company" element="tns:QuoteCompany" />
</message>
- <portType name="ptProvideQuote">
  <operation name="xmlProvideQuote">
```

The SeeBeyond UDDI Registry can be used in a third party tool, for example Microsoft Visual Studio (see Figure 151). In Visual Studio's Solution Explorer, right-click on *References* and enter the URL of the SeeBeyond Web Services page as a Web Reference.

**Figure 151** Microsoft Visual Studio Example



eInsight Enterprise Service Bus can exchange data over the Internet with Web Services applications described using the Web Services Description Language (WSDL). This language 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 112).

### 9.3.1 Using UDDI Browsers

A third-party UDDI Browser can be used to access the SeeBeyond UDDI Registry for publishing and inquiry. To add the SeeBeyond UDDI Registry to the browser, you need to enter the information given in Table 60 into the browser's *Add UDDI Registry* facility.

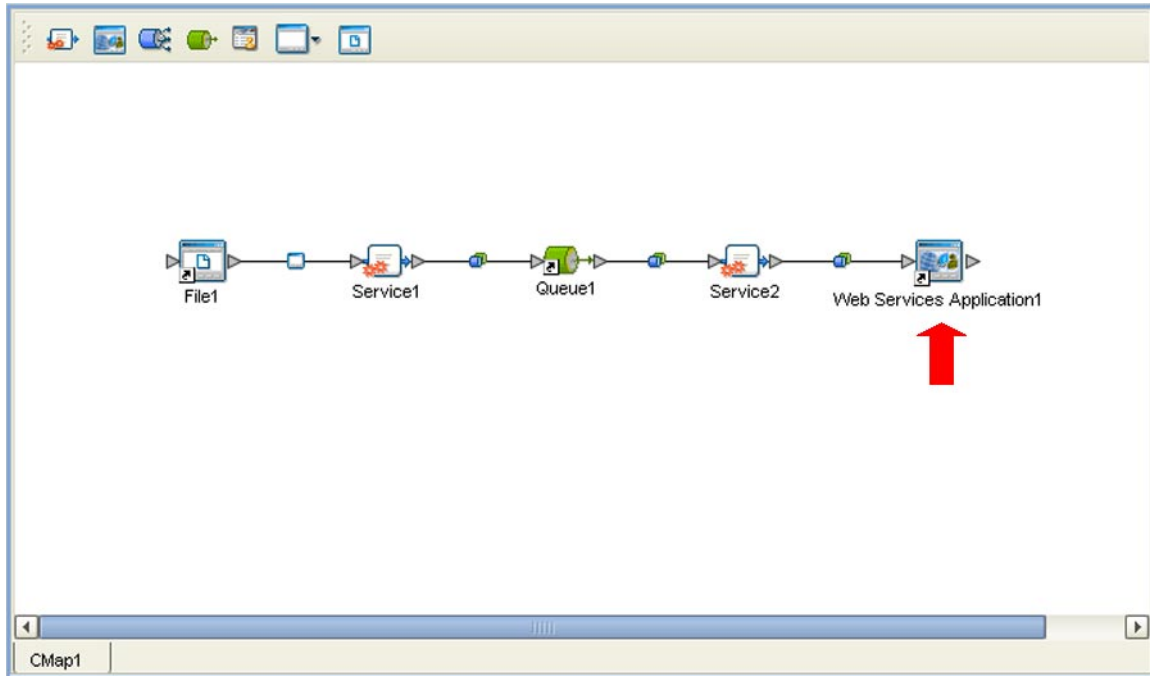
**Table 60** UDDI Registry Information

Parameter	Value
Name	SeeBeyond UDDI
Inquiry URL	http:// <i>hostname:portnumber</i> /stcuddi/inquiry
Publish URL	http:// <i>hostname:portnumber</i> /stcuddi/publish
Username	<i>your_username</i>
Password	<i>your_password</i>

## 9.4 Web Services Application

A Web Services Application (see Figure 152) represents either an external Web Service that can be invoked by a Web Services Client, based on an eInsight Business Process, or a Business Process exposed as a Web Service. In the case of the latter, the Web Service is listed in the SeeBeyond UDDI Registry.

**Figure 152** Enterprise Designer - Connectivity Map Editor

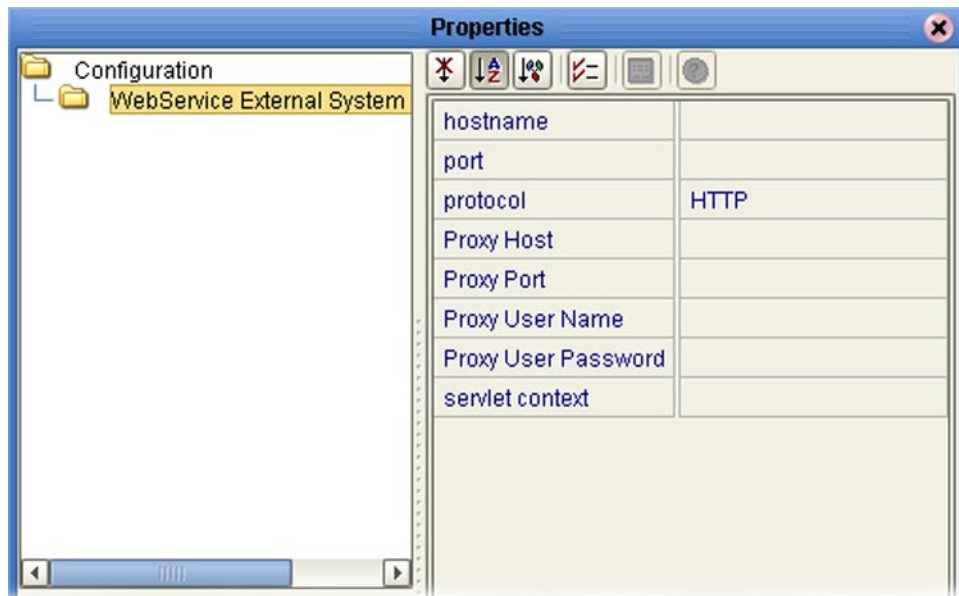


If you are creating more than one Web Service Application for deployment in any given Environment, you must ensure that the respective *servlet context* properties have different values so that each application will have a different URL. See [Web Services External System](#) on page 197

## 9.5 Web Services External System

A Web Services External System represents a system containing a Web Services Application. The Properties dialog box (see Figure 153) appears automatically when you create a New Web Services External System in the Environment Explorer. You can also invoke the dialog box by right-clicking the Web Services Application in the Enterprise Explorer and selecting **Properties**. You should assign the desired values for the listed properties.

**Figure 153** Web Service External System Properties Dialog Box



**Table 61** Web Services External System Properties

Property	Description
hostname	Name or IP address of the computer hosting the application. There is no default.
port	Port on which the application can be accessed. There is no default.
protocol	WSDL end point protocol ( <b>HTTP</b> or <b>HTTPS</b> ). If SSL is enabled for the Web Container (see <a href="#">Default Web Server</a> on page 142), this parameter must be set to HTTPS prior to deployment. The default is <b>HTTP</b> .
Proxy Host	Name of the proxy host, if any. There is no default.
Proxy Port	Port for the proxy host, if any. There is no default.
Proxy User Name	User name for the proxy host, if any. There is no default.
Proxy User Password	User password for the proxy host, if any. There is no default.
servlet context	The path and name of the Web Services Application.

If you do not specify the properties for the Web Service, then they are assigned automatically upon deployment. The name automatically assigned to the **servlet context** property can be lengthy and complex, however, so you should rename it to something relatively simple. This property controls the name of the Web Service Application (**.war**) file.

Additionally, if you are creating more than one Web Service Application for deployment in any given Environment, you must ensure that the respective servlet context properties have different values so that each application will have a different URL. The URL format is **http://hostname:port/servlet context**.

**Note:** *You must specify the hostname and port if the Project is not deployed on the local system.*

**Important:** *If you are using the HTTPS protocol, the value of the hostname property must exactly match that used to create the certificates.*

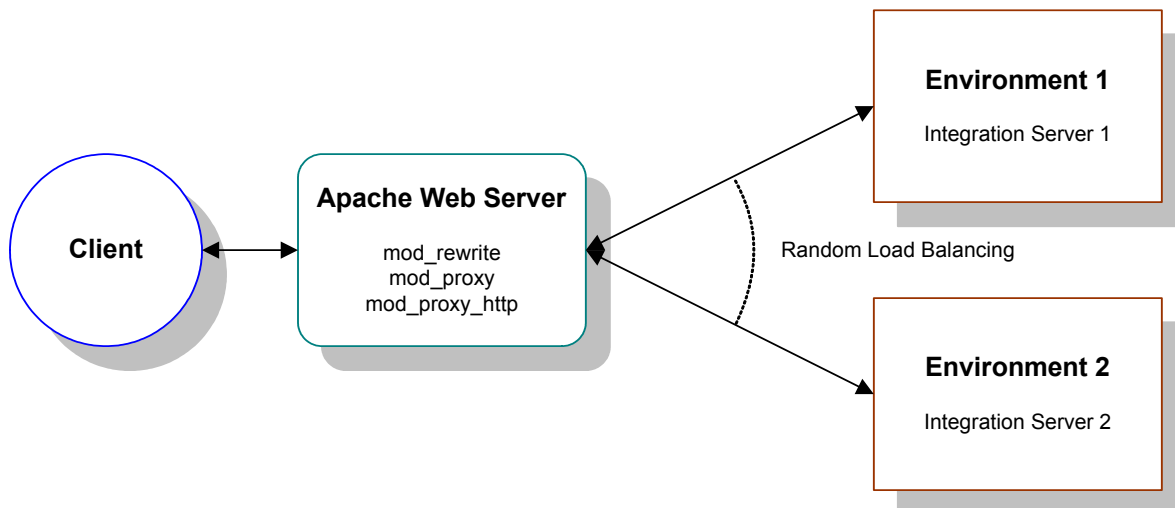
## 9.6 Load Balancing

You can build a system that will load balance HTTP requests in a way suitable for ICAN Web Services by using an Apache-based Web server. To accomplish this you will need the following:

- 1 An Apache-based HTTP Web server such as Apache 2.0 or IBM HTTP 2.0.
- 2 Modify the Apache configuration file to enable **mod\_rewrite**, **mod\_proxy**, and **mod\_proxy\_http**.
- 3 Create a mapping file that includes the name of your business process that is published as a Web Service and a list of URL endpoints where it is available.
- 4 Add the **mod\_rewrite** rewrite rule to your Apache configuration file.

**Note:** *This scheme only provides simple random load-balancing. It does not provide fail-over nor does it provide any heuristics or any other ability to distribute work based on server load.*

**Figure 154** Load Balancing Example



### 9.6.1 Configuring the Apache Server

The main configuration file for Apache, **httpd.conf**, is located in **APACHE/config**. To enable **mod\_proxy**, **mod\_proxy\_http**, and **mod\_rewrite**, uncomment the following lines shown in bold text (these options are usually disabled—commented out—by default).

```
#LoadModule mime_magic_module modules/mod_mime_magic.so
LoadModule proxy_module modules/mod_proxy.so
#LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so
#LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule setenvif_module modules/mod_setenvif.so
```

## 9.6.2 Creating the Mapping File

Before you create the Web Services and the mapping file you need to be aware of the mapping rule that will be used. The URL generated for the SeeBeyond Web Services endpoint is of the following form (you must rename the *bold\_italic* section in the servlet context property as described in [Web Services External System](#) on page 197):

```
WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/STCService
```

The mapping file (**ws\_server.txt**) uses the business process name as the key to find a list of URL endpoints on which that service will be available. This file is a simple text file of the following form:

```
Key alternate_URL1|alternate_URL2
```

The content below is a typical example—however, it should actually appear on a single line with no space between the alternate URLs. The alternate URLs contain the actual endpoints for the Web Services, including the host IP address and port.

```
bpCalc1
127.0.0.1:18004/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/STCService|
127.0.0.1:19004/WSServlet_WSCalc_u02F_IS2_WSC_WSCALC_bpCalc1/STCService
```

The mapping file must be stored on the same system that is running Apache, and is usually found in the configuration directory, for example:

```
C:\Program Files\IBM HTTP Server 2.0\conf\ws_server.txt.
```

## 9.6.3 Adding the Rewrite Rule

The code section below is the rewrite rule; lines beginning with '#' are comments:

```
# Start of ICAN rewrite rules
<IfModule mod_rewrite.c>
#RewriteLog "C:\Program Files\IBM HTTP Server 2.0\logs\rewrite.log"
#RewriteLogLevel 5
RewriteMap SERVICES "rnd:C:\Program Files\IBM HTTP Server
2.0\conf\ws_server.txt"

<Location ~ /WSServlet_ >
    RewriteEngine On

    RewriteRule "([^\_]+(?!/STCService))" "http://${SERVICES:$1}" [P,L]
    RewriteRule "(.*)" "http://localhost:80/notfound.html" [G]
</Location>
</IfModule>
# End of rewrite directives.
```

The purpose of this rule is to intercept requests for ICAN-formatted URLs of the form

```
http://LOADBALANCER:9999/WSServlet_XXXXXXX_bpCalc1/STCService
```

and rewrite them to a form using the entries in the mapping file **ws\_server.txt**. It does this by extracting the key (shown in bold) from the requested URL. This key is the last element of the variable part of the endpoint URL created by ICAN.

- If it cannot find a suitable match, it will return a *gone* (410) error.
- If it finds a suitable key but does not find a match in the mapping file, it will return a *not found* (404) error.
- If a key is matched but the service is not available on that URL, it will return a *bad gateway* (502) error.



## 9.6.4 Debugging

There are a number of ways to debug the internal processing of Apache, the rewrite logging probably being the most useful. You can specify an additional rewrite log and a log level within the rewrite section of the **httpd.conf** file. The logging level is on a scale of 0 to 5, with 0=off and 5=maximum debugging.

An example of a successful rewrite request log is shown below, using log level 5 (timestamps omitted):

```
[rid#596d60/initial] (3) [per-dir /WSServlet_/] add path info postfix: C:/
Program Files/IBM HTTP Server 2.0/htdocs/en_US/WSServlet_WSCalc_
u02F_IS1_WSC_WSCALC_bpCalc1 -> C:/Program Files/IBM HTTP Server 2.0/
htdocs/en_US/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/STCService

[rid#596d60/initial] (3) [per-dir /WSServlet_/] applying pattern '([^/_]+(?=/
STCService))' to uri 'C:/Program Files/IBM HTTP Server 2.0/htdocs/en_US/
WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/STCService'

[rid#596d60/initial] (5) cache lookup OK: map=SERVICES[txt] key=bpCalc1 ->
val=127.0.0.1:18004/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/
STCService

[rid#596d60/initial] (5) randomly chosen the subvalue `127.0.0.1:18004/
WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/STCService'

[rid#596d60/initial] (2) [per-dir /WSServlet_/] rewrite C:/Program Files/IBM
HTTP Server 2.0/htdocs/en_US/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_
bpCalc1/STCService -> http://127.0.0.1:18004/WSServlet_WSCalc_u002F_
IS1_WSC_WSCALC_bpCalc1/STCService

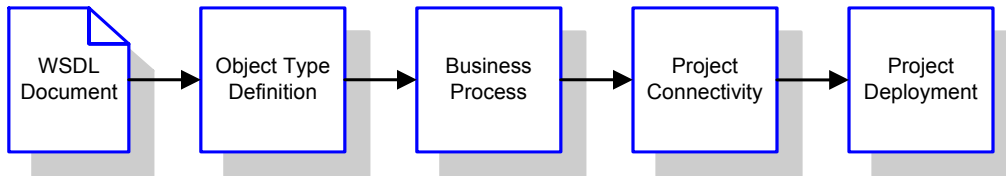
[rid#596d60/initial] (2) [per-dir /WSServlet_/] forcing proxy-throughput with
http://127.0.0.1:18004/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_bpCalc1/
STCService

[rid#596d60/initial] (1) [per-dir /WSServlet_/] go-ahead with proxy request
proxy:http://127.0.0.1:18004/WSServlet_WSCalc_u02F_IS1_WSC_WSCALC_
bpCalc1/STCService [OK]
```

## 9.7 Building a Web Service

A simplified process flow for developing a Web Service is shown in Figure 155.

**Figure 155** Development Flow



### 9.7.1 Procedure

#### To create a Web Service

- 1 Create the Web Service Project (see [Creating a Project](#) on page 84).
- 2 Add a new Object Type Definition (OTD) to the Project.
- 3 Build the OTD using the WSDL OTD Wizard, based on an externally-provided WSDL file.
- 4 Add a New Business Process to the Project.
- 5 Develop the Business Process in eInsight Business Process Manager, based on the OTD.
- 6 Map the Project components.
- 7 Deploy the Project to the selected Environment.

## 9.7.2 Web Server Example

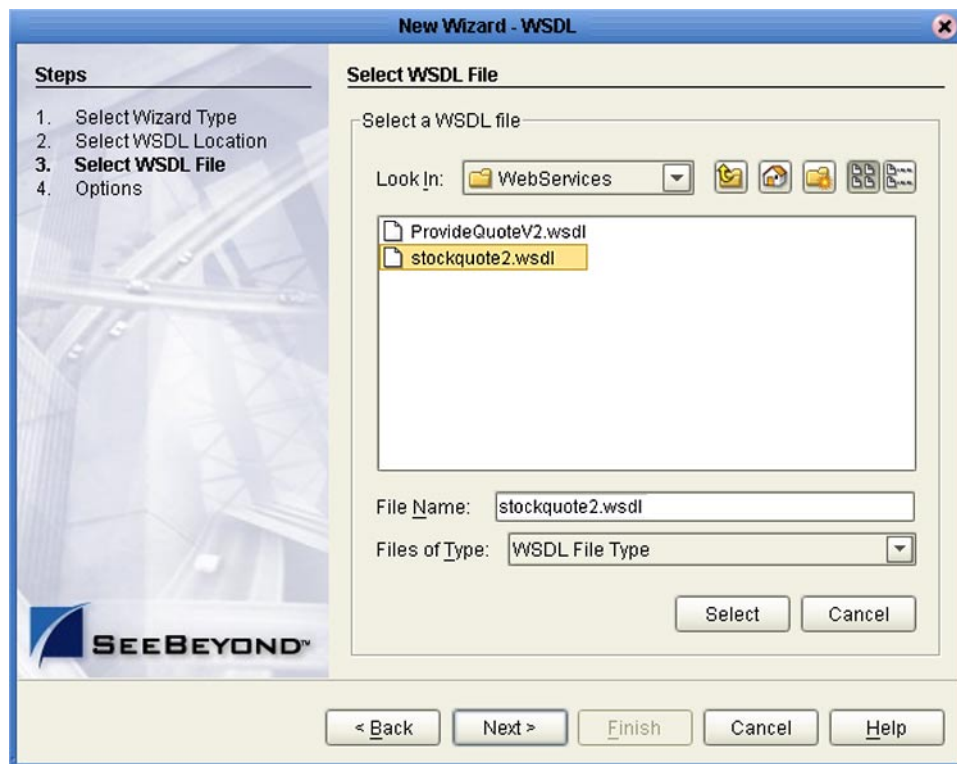
The Project used in the following example is contained in the *Web Services Sample* listed on the Enterprise Manager’s Documentation page. To use this example Project, download the sample file **Web\_Services\_Sample.zip** and extract the contents to a convenient directory. Import the file **webserver.zip** into your Repository following the procedure described in [Project/Environment Import](#) on page 53.

**Note:** For the example Project to work correctly, you must first create and deploy your Web Service, then create and deploy the Web Services Client—deploying both to the same Web Service External System.

### Object Type Definition

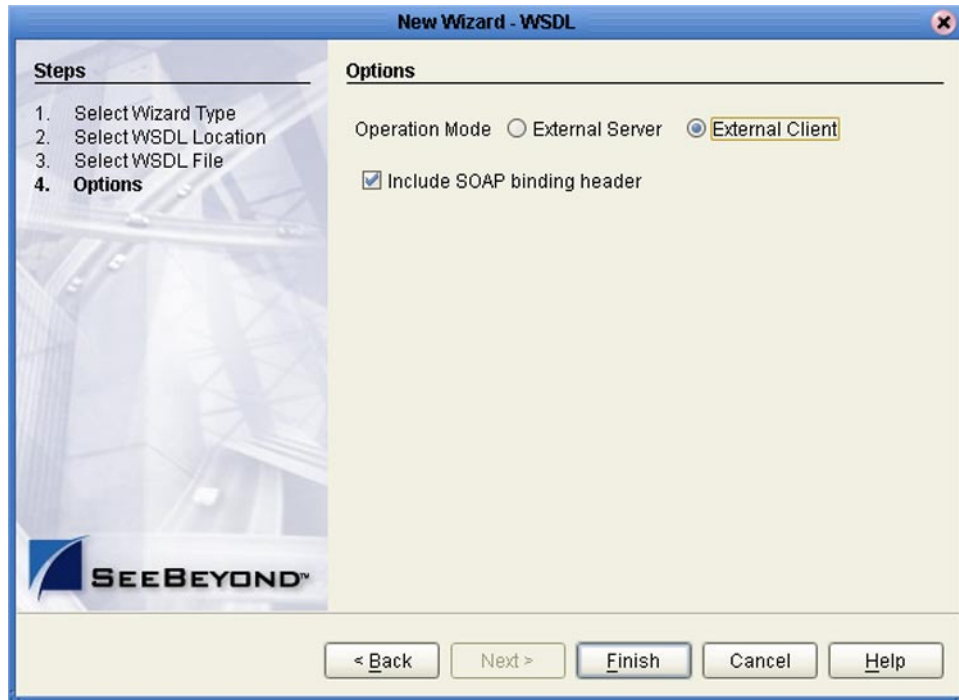
The OTD for the sample Web Service was created as described in [Using the WSDL Wizard](#) on page 112. The WSDL file must be provided by an external source; the file used in the example is shown in Figure 156.

**Figure 156** Wizard Step 3 - Select WSDL File



In the *Options* step, the Operational Mode for the Web Service is **External Client** (see Figure 157).

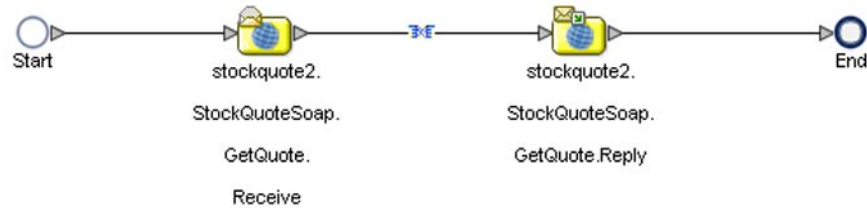
Figure 157 Wizard Step 4 - Options



## eInsight Business Process

The example Business Process is shown in Figure 158 (see the *eInsight Enterprise Service Bus User's Guide* for details).

Figure 158 Web Service Business Process



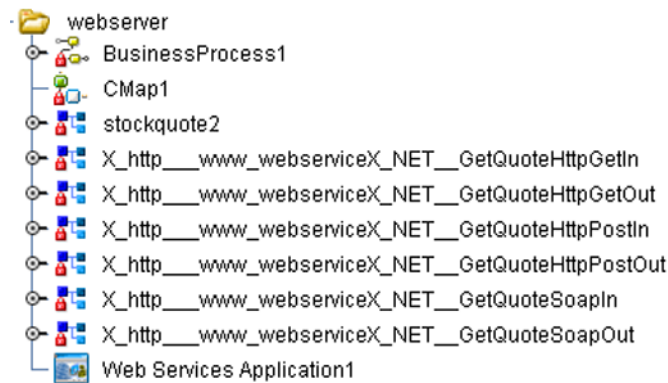
## Project Mapping

The Project components are created and connected in the Enterprise Designer Connectivity Map Editor. The example Project contains:

- An External Web Services Application.
- A service, into which you drag and drop the eInsight Business Process from the Project Explorer.

The Web server example Project appears in the Project Explorer as shown in Figure 159

**Figure 159** Web Service Example Project - Explorer Tree



The Business Process is connected as shown in Figure 160.

**Figure 160** Web Service Example Project - Business Process Mapping



Collapsing the binding box results in Figure 161.

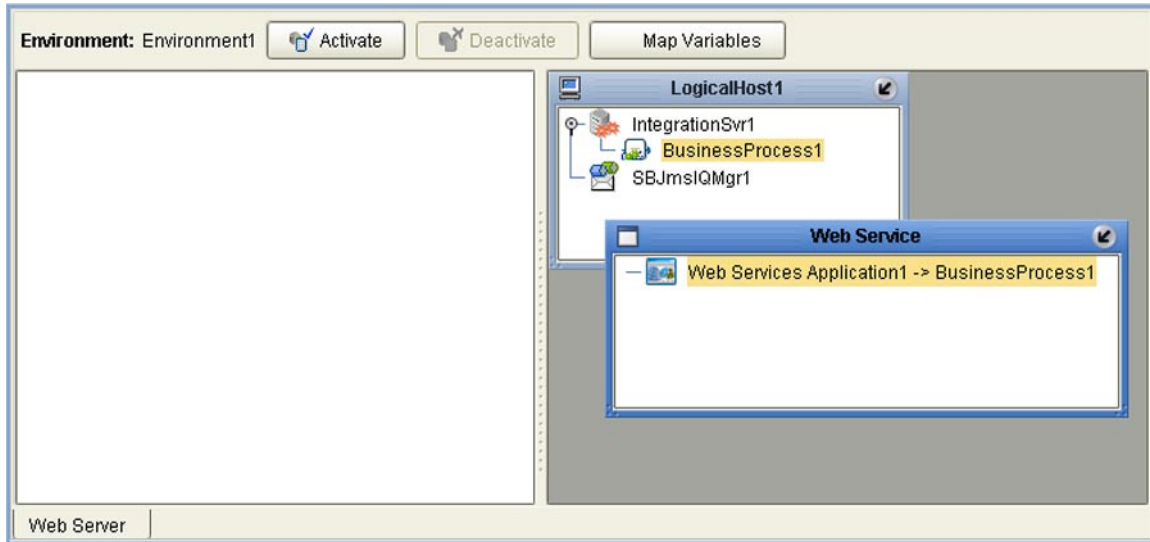
**Figure 161** Web Service Example Project - Connectivity Map



## Project Deployment

The Project is deployed to a run-time Environment as shown in Figure 162. (See [Creating a Deployment Profile](#) on page 176 and [Activating and Deactivating Deployment Profiles](#) on page 182 for information.)

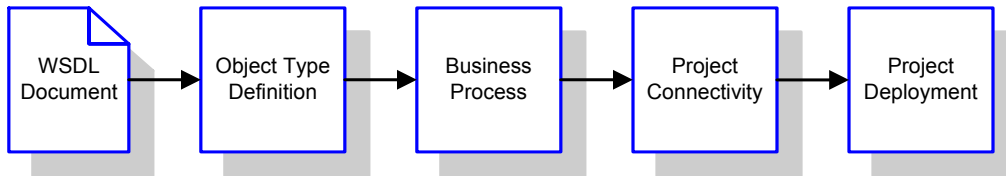
**Figure 162** Web Server Example Project - Deployment



## 9.8 Building a Web Services Client

A simplified process flow for developing a Web Services Client is shown in Figure 163.

**Figure 163** Development Flow



### 9.8.1 Procedure

#### To build a Web Services Client

- 1 Create the Web Services Client Project (see [Creating a Project](#) on page 84).
- 2 Add a new Object Type Definition (OTD) to the Project.
- 3 Build the OTD using the WSDL OTD Wizard, based on an externally-provided WSDL file.
- 4 Add a New Business Process to the Project.
- 5 Develop the Business Process in eInsight Business Process Manager, based on the OTD.
- 6 Map the Project components.
- 7 Deploy the Project to the selected Environment.

## 9.8.2 Web Client Example

The Project used in the following example is contained in the *Web Services Sample* listed on the Enterprise Manager’s Documentation page. To use this example Project, download the sample file **Web\_Services\_Sample.zip** and extract the contents to a convenient directory. Import the file **webclient.zip** into your Repository following the procedure described in **Project/Environment Import** on page 53. The files **input.txt** and **output.txt** are simple text files for testing purposes.

**Note:** For the example Project to work correctly, you must first create and deploy your Web Service, then create and deploy the Web Services Client—deploying both to the same Web Service External System.

### Object Type Definition

The OTD for the sample Web Services Client was created as described in **Using the WSDL Wizard** on page 112. The required WSDL file (see Figure 164) is produced by the Web Service developed in the preceding section, and published in the UDDI registry upon deployment and activation.

**Figure 164** Example Web Service WSDL File

```
<?xml version="1.0" encoding="UTF-8" ?>
- <definitions targetNamespace="http://seebeyond/quoteservice" xmlns:tns="http://seebeyond/quoteservice"
  xmlns:sbypnx="http://bpel.seebeyond.com/hawaii/5.0/privateExtension/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
- <types targetNamespace="http://seebeyond/quoteservice">
  - <xsd:schema targetNamespace="http://seebeyond/quoteservice"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element type="xsd:string" name="QuoteRequest" />
    <xsd:element type="xsd:string" name="QuotePrice" />
    <xsd:element type="xsd:string" name="QuoteCompany" />
    </xsd:schema>
  </types>
- <message name="msgProvideQuoteRequest">
  <part name="TICKER" element="tns:QuoteRequest" />
</message>
- <message name="msgQuoteResponse">
  <part name="price" element="tns:QuotePrice" />
  <part name="company" element="tns:QuoteCompany" />
</message>
- <portType name="ptProvideQuote">
  <operation name="provideQuote">
```

The file location is most easily specified by locating the file using the UDDI browser (see Figure 165), and entering its URL in the WSDL Wizard as shown in Figure 166.

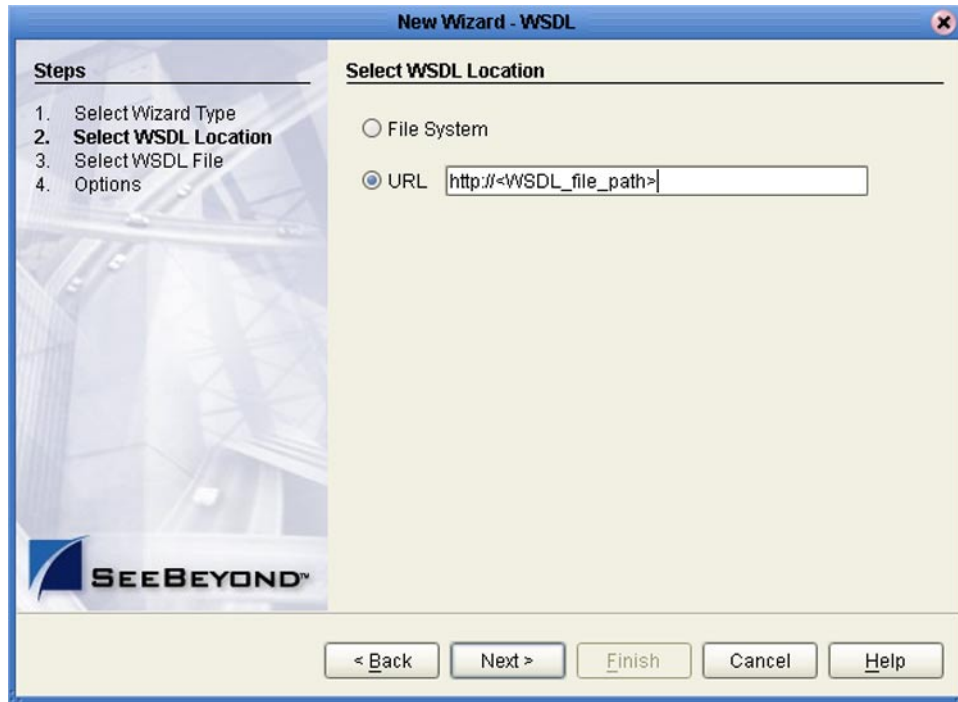
**Figure 165** SeeBeyond UDDI Browser



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

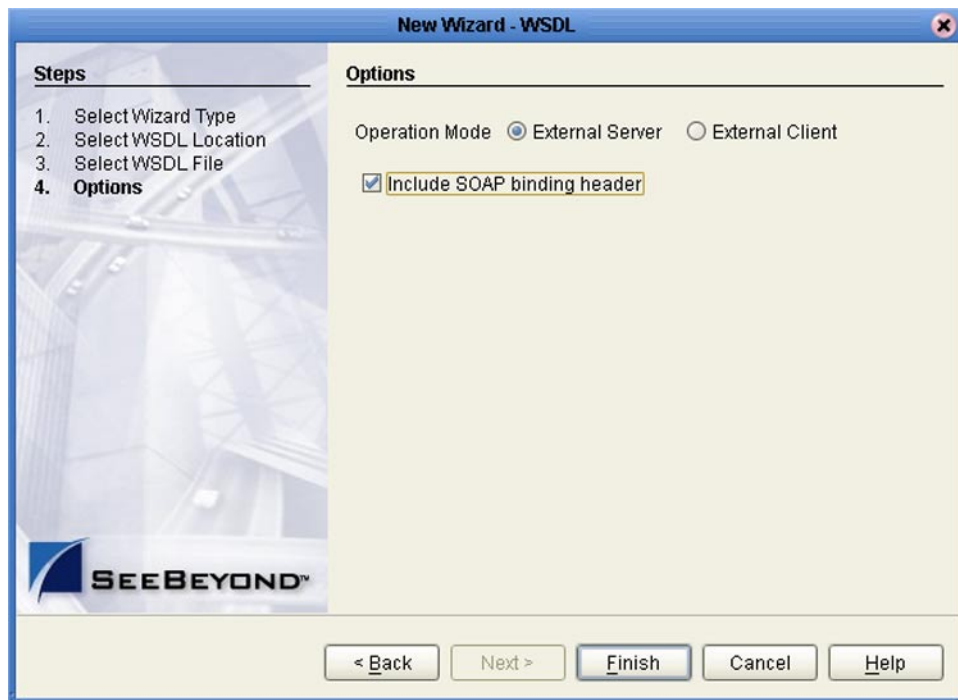


Figure 166 Select WSDL File



In the *Options* Wizard, the Operational Mode for the Web Services Client is **External Server** (see Figure 167).

Figure 167 Wizard Step 4 - Options



## eInsight Business Process

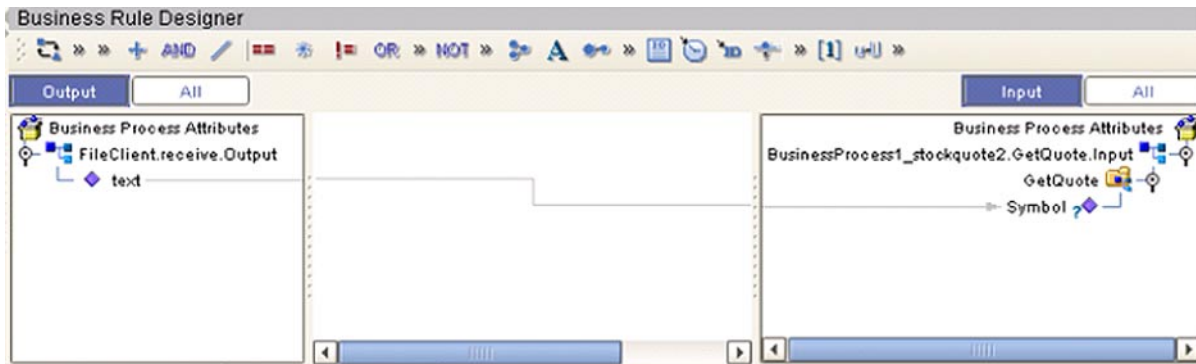
The example Business Process is shown in Figure 168 (see the *eInsight Enterprise Service Bus User's Guide* for details).

**Figure 168** Web Client Business Process

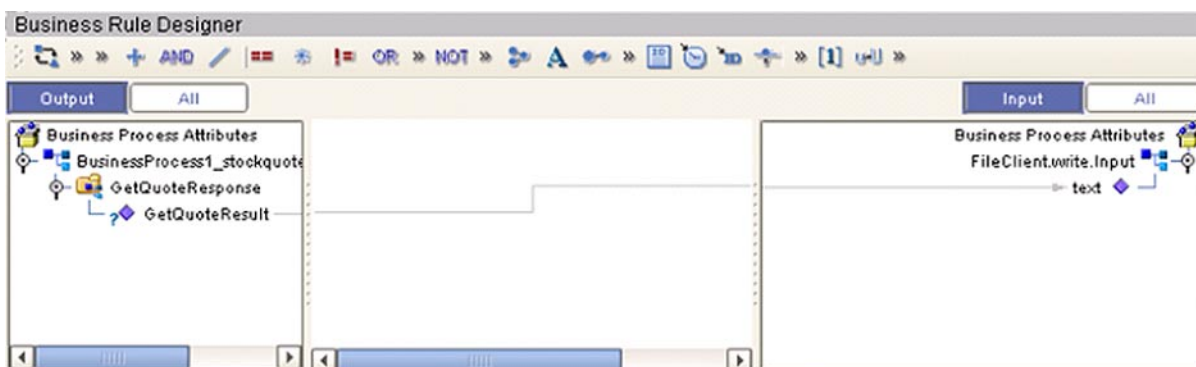


The **receive** rule for the Business Process is shown in Figure 169, and the **write** rule is shown in Figure 170.

**Figure 169** Web Client Business Process Receive Rule



**Figure 170** Web Client Business Process Write Rule



## Project Mapping

The Project components are created and mapped in the Enterprise Designer Connectivity Map Editor. The example Project contains:

- Two external files and accompanying File eWays.
- An External Web Service.
- A service, into which you drag and drop the eInsight Business Process from the Project Explorer.

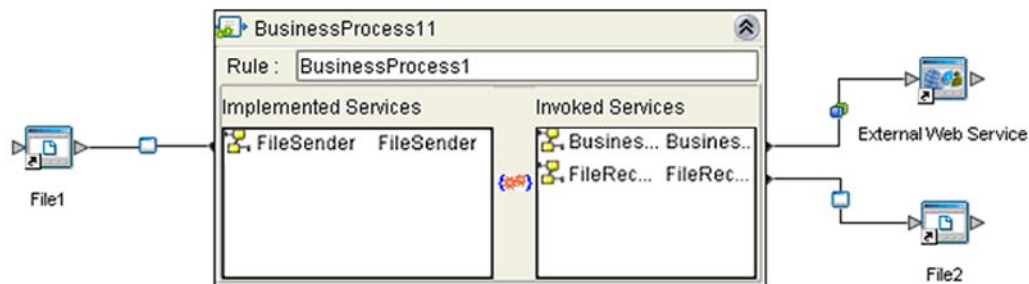
The Web Services Client example Project appears in the Project Explorer as shown in Figure 171.

**Figure 171** Web Services Client Example Project - Explorer Tree



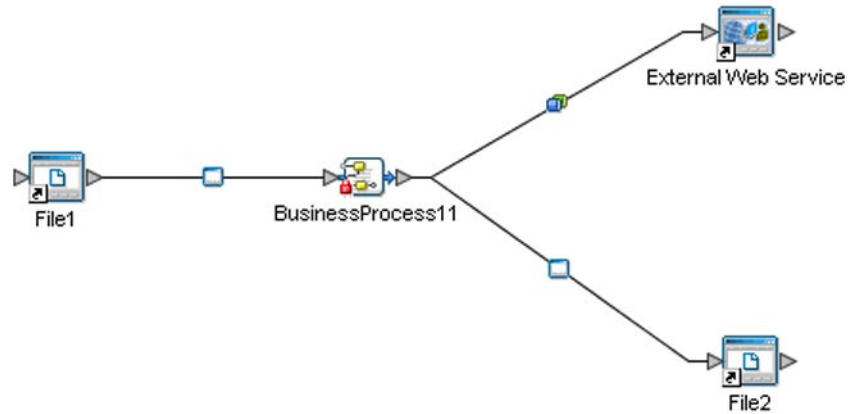
The Business Process is connected as shown in Figure 172.

**Figure 172** Web Services Client Example Project - Business Process Mapping



Collapsing the binding box results in Figure 173.

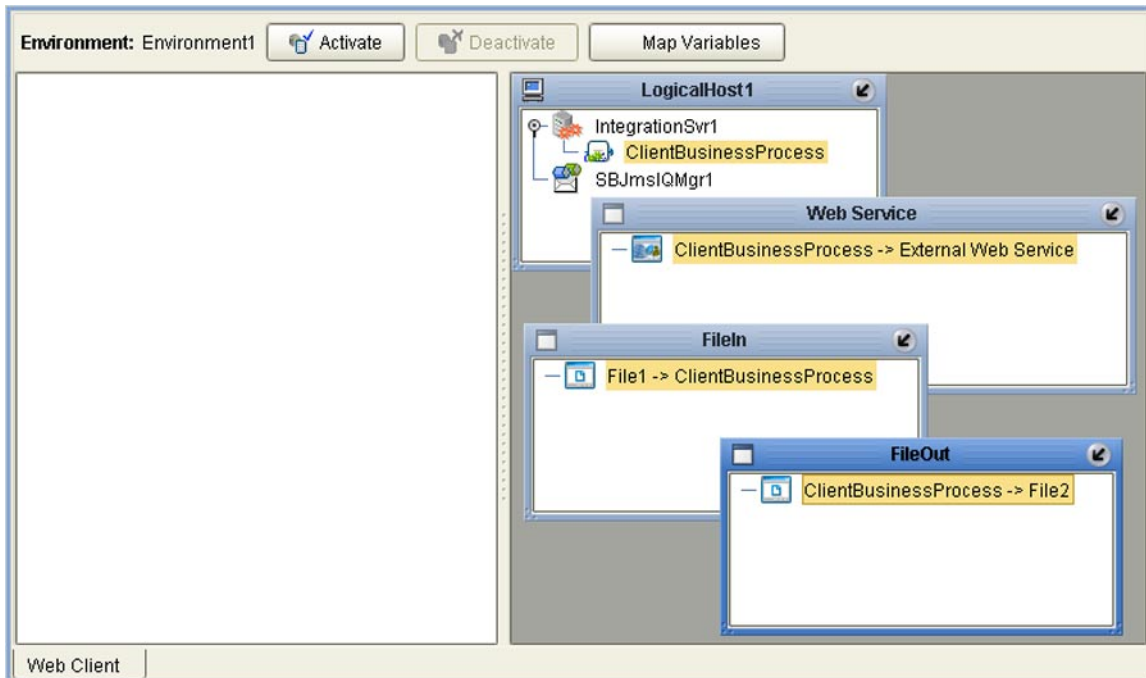
**Figure 173** Web Services Client Example Project - Connectivity Map



## Project Deployment

The example Project is deployed to a run-time Environment as shown in Figure 174. (See [Creating a Deployment Profile](#) on page 176 and [Activating and Deactivating Deployment Profiles](#) on page 182 for information.)

**Figure 174** Web Services Client Example Project - Deployment



If the Web Service and Web Services Client are deployed to different Web Services External Systems (usually in different Environments), you must manually copy the configuration from the Web Services External System in which the Service is deployed to the one in which the Client is deployed. See [Web Services External System](#) on page 197.

## Project Operation

### To run the Web Services Example Project

- 1 Start the Logical Host (see [Bootstrapping](#) on page 135).
- 2 Copy the file **input.txt** into the directory that the **File1** eWay polls for input files (**C:\temp**).
- 3 A valid output file (see **output.txt**) should appear in the directory into which the **File2** eWay places files (**C:\temp**).

---

## 9.9 Setting Up Secure Web Services

### 9.9.1 SSL Overview

Secure Sockets Layer (SSL) 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, and *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, the 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 to verify that the client is who and what it claims to be. This 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, and so on), indicating 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 that the data indeed comes from whoever claims to have created and signed it.

The certificate is cryptographically signed by its originator 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. For testing purposes or use within an adequately controlled environment, a self-signed certificate can be used instead.

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

## Keytool Utility

One tool that can be used to set up a server or client 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 or herself to other users or 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.

The keys and certificates are stored in a *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.

**Note:** *If the Web Service and Web Services Client are deployed to different Web Services External Systems (usually in different Environments), you must manually copy the configuration from the Web Services External System in which the Service is deployed to the one in which the client is deployed. See [Web Services External System](#) on page 197.*

## 9.9.2 Web Services Setup

Setup procedures for using SSL are described in detail in **Configuring SSL Support** in the *eGate Integrator System Administration Guide*. A brief outline of these procedures is included in the following.

**Note:** *You must first create and deploy your Web Service, then create and deploy the Web Services Client—deploying both to the same Web Service External System.*

### Web Service Setup

To set up SSL for a Web Service

- 1 Create the Web Service as described in **Building a Web Service** on page 202.
- 2 Configure your Integration Server to use SSL.
  - A Configure the Web Container (see **Web Container Configuration** on page 141 and **Default Web Server** on page 142). If you have already deployed non-SSL Web Services Projects using properties defined in *Web Container Configuration: Default Web Server*, then you must create a new set of configuration properties as described in **Web Server Configurations** on page 141, enabling SSL and using a different Connector Port.
  - B Create a new Web Services External System (see **Web Services External System** on page 197), setting the **protocol** property to **HTTPS**. The **port** number must match the **Connector Port** in the corresponding Web Server, as set in step A above.

**Note:** *If you need a self-signed Certificate, skip step C and create the server Certificate and Keystore using the script in **Creating Keystores and Self-Signed Certificates** on page 224. Then continue with step D.*

- C Right-click the Environment and create a new **Keystore**.
    - D Right-click the Keystore and select **Manage Trust Stores**.
    - E Click **New** and give the Truststore an alias name. The *Certificate* fields will be enabled.
    - F Import the server Certificate into the Truststore.
- 3 Create a new Deployment Profile (see **Creating a Deployment Profile** on page 176).
- 4 Start the Logical Host (see **Bootstrapping** on page 135), which creates a new **keystore** directory under *ICAN-root\logicalhost*.
- 5 Copy the file **IntegrationSvr1.keystore** into this directory, which will allow the Web Container to find the SSL certificate.
- 6 Activate the Web Service Project (see **Activating and Deactivating Deployment Profiles** on page 182), which should restart the Logical Host.



## Web Services Client Setup

### To set up SSL for a Web Services Client

- 1 Create the Web Services Client as described in [Building a Web Services Client](#) on page 207.
- 2 Import the server Certificate into the client Truststore using the command line:
  - A For design time, use the command:

```
Keytool -keystore cacerts -alias aliasname -import - file server.cer  
ICAN-root(client)\edesigner\jdk\jre\lib\security
```
  - B For run time, use the command:

```
Keytool -keystore cacerts -alias aliasname -import - file server.cer  
ICAN-root(client)\logicalhost\jre\lib\security
```
- 3 Create a new Deployment Profile, deploying the Web Services Client to the same Web Services External System as the Web Service (see [Creating a Deployment Profile](#) on page 176).
- 4 Activate the Project (see [Activating and Deactivating Deployment Profiles](#) on page 182).
- 5 If the Logical Host is running, restart it (see [Bootstrapping](#) on page 135).

### 9.9.3 Web Services Example Project

An example Web Services Project is contained in the *Web Services SSL Sample* listed on the Enterprise Manager's Documentation page. To use this example Project, download the sample file **Web\_Services\_SSL\_Sample.zip** and extract the contents to a convenient directory.

To make the example Project self-contained, self-signed Certificates are used for authentication. A script is provided in a batch file to generate the required Keystores and Certificates, and import the Certificates into the appropriate Keystores. The example does not incorporate encryption.

**Note:** For the example Project to work correctly, you must first create and deploy your Web Service, then create and deploy the Web Services Client—deploying both to the same Web Service External System.

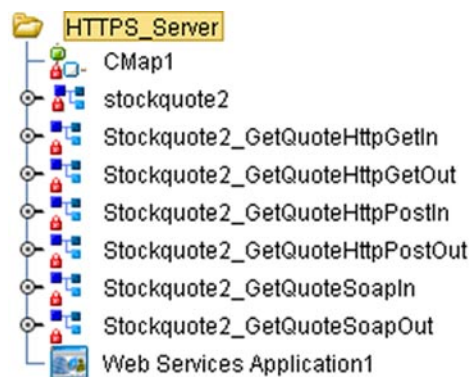
## Web Service Example

**Note:** You must have eInsight installed in your ICAN system.

To implement the example Web Service

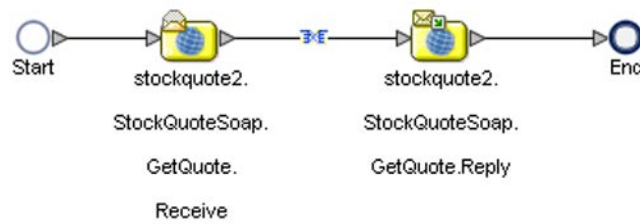
- 1 Import the file **HTTPS\_Server.zip** into your Repository following the procedure described in **Project/Environment Import** on page 53. The Project should appear in the Project Explorer as shown in Figure 175.

**Figure 175** Example Secure Web Services Project - Explorer Tree



- 2 The Business Process should appear as shown in Figure 176.

**Figure 176** Example Secure Web Services Project - Business Process



- 3 The Project Connectivity Map should appear as shown in Figure 177.

**Figure 177** Example Secure Web Services Project - Connectivity Map



- 4 Configure the Default Web Server to use SSL (see [Default Web Server](#) on page 142).
  - A *Enable SSL*: **true**
  - B *SSL Keystore Password*: **tomcat**
  - C *Connector Port*: *(default)*

**Note:** *If you have already deployed non-SSL Web Services Projects using properties defined in Web Container Configuration: Default Web Server, then you must create a new set of configuration properties as described in [Web Server Configurations](#) on page 141, enabling SSL and using a different Connector Port.*

- 5 Create a new Web Services External System (see [Web Services External System](#) on page 197).
  - A *hostname*: the IP address or domain name where the Logical Host will be running.
  - B *port*: the same as the **Connector Port** specified in step 4 above.
  - C *protocol*: **HTTPS**
- 6 Create the server Keystore and the required self-signed Certificates (see [Creating Keystores and Self-Signed Certificates](#) on page 224).
- 7 Right-click the Keystore and select **Manage Trust Stores**.
- 8 Click **New** and give the store an alias name. The Certificate fields will be enabled.
- 9 Import the server Certificate into the Truststore.

- 10 Create a new Deployment Profile (see [Creating a Deployment Profile](#) on page 176).
- 11 Start the Logical Host (see [Bootstrapping](#) on page 135), which creates a new **keystore** directory under *ICAN-root\logicalhost*.
- 12 Copy your **IntegrationSvr1.keystore** file into this directory, which will allow the Web Container to find the SSL certificate.
- 13 Activate the Web Service Project (see [Activating and Deactivating Deployment Profiles](#) on page 182), which should restart the Logical Host.

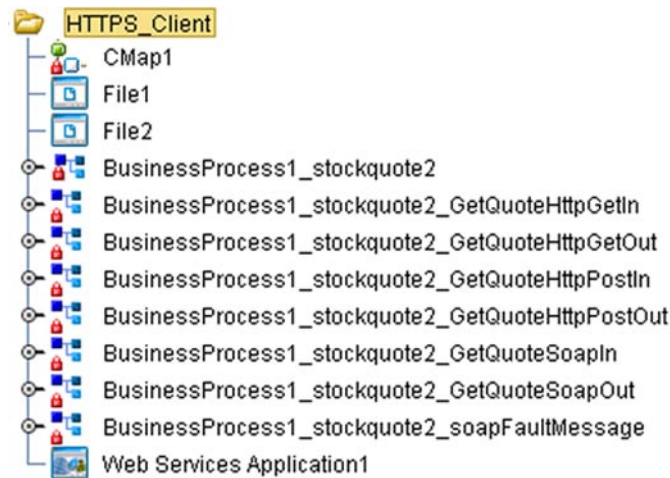
## Web Services Client Example

**Note:** You must have eInsight installed in your ICAN system.

To implement the example Web Services Client

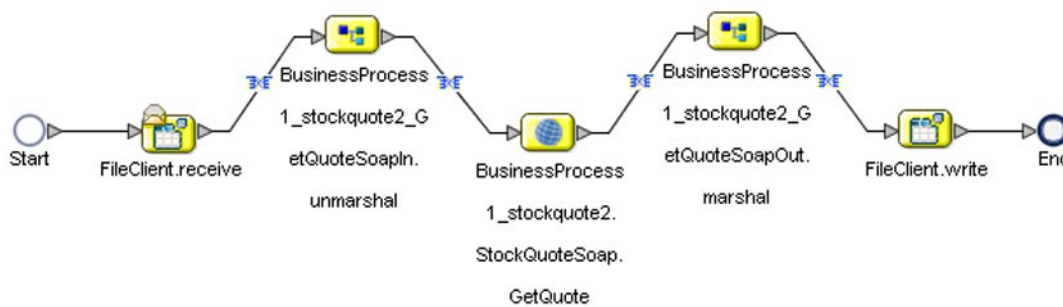
- 1 Import the file **HTTPS\_Client.zip** into your Repository following the procedure described in **Project/Environment Import** on page 53. The Project should appear in the Project Explorer as shown in Figure 178.

**Figure 178** Example Secure Web Services Client Project - Explorer Tree



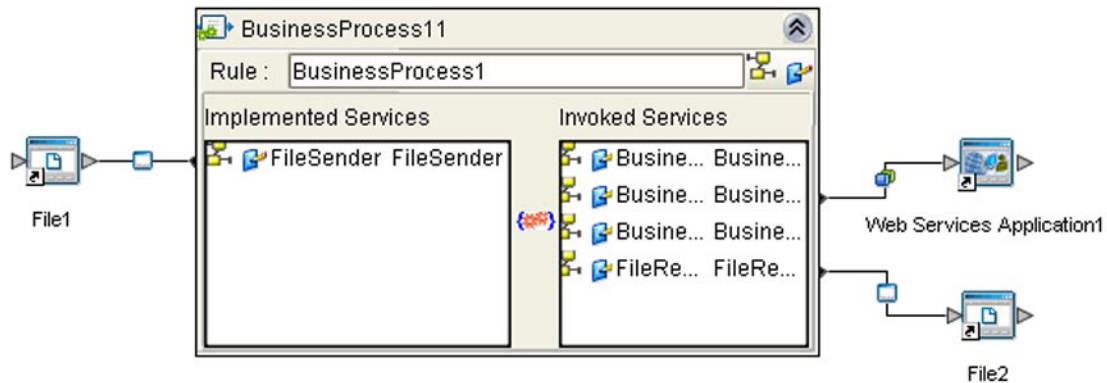
- 2 The Business Process should appear as shown in Figure 179.

**Figure 179** Example Secure Web Services Client Project - Business Process

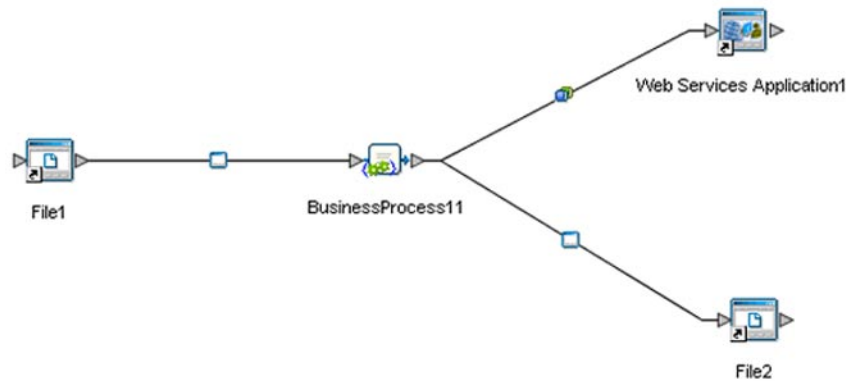


- 3 The Project Connectivity Map should appear as shown in Figure 180 and Figure 181.

**Figure 180** Example Secure Web Services Client Project - Business Process Mapping



**Figure 181** Example Secure Web Services Client Project - Connectivity Map



- 4 Open a Command Prompt window and import the server Certificate (server.cer) into the client Truststore (tomcat-cl).
  - A For design time, use the command:  

```
Keytool -keystore cacerts -alias tomcat-cl -import - file server.cer
ICAN-root(client)\edesigner\jdk\jre\lib\security
```
  - B For run time, use the command:  

```
Keytool -keystore cacerts -alias tomcat-cl -import - file server.cer
ICAN-root(client)\logicalhost\jre\lib\security
```

**Note:** When asked, use the default password *changeit*.

- 5 Create a new Deployment Profile, deploying the Web Services Client to the same Web Services External System as the Web Service (see [Creating a Deployment Profile](#) on page 176).
- 6 Activate the Web Services Client Project (see [Activating and Deactivating Deployment Profiles](#) on page 182).
- 7 If the Logical Host is running, restart it (see [Bootstrapping](#) on page 135).

## Project Operation

### To run the Web Services Example Project

- 1 Start the Logical Host, if it is not already running (see [Bootstrapping](#) on page 135).
- 2 Copy the file **inputssl.in** into the directory that the **File1** eWay polls for input files (**C:\temp**).
- 3 A valid output file (see **outputssl1.dat**) should appear in the directory into which the **File2** eWay places files (**C:\temp**).

## 9.9.4 Creating Keystores and Self-Signed Certificates

The sample file **Web\_Services\_SSL\_Sample.zip** contains a batch file (**key\_cert.bat**) containing the following script for use with the example Project. This script will:

- Generate the server and client Keystores.
- Create a self-signed certificate for the server.

The parameter **<servername>** must be a fully-qualified domain name, and must match the **hostname** in the Web Services External System configuration. The default password for the certificates is **tomcat**, which you can change if desired.

You should set the environmental variable **JAVA\_HOME** to point to your JDK installation (**ICAN\_root\repository\jre\1.4.2\_04**).

### Example Project Script

```
if not "%JAVA_HOME%" == "" goto gotJavaHome
goto cleanup
:gotJavaHome

echo Generating the Server Keystore in file IntegrationSvr1.keystore
%java_home%\bin\keytool -genkey -alias tomcat-sv -dname
    "CN=<servername>, OU=X, O=Y, L=Z, S=XY, C=YZ" -keyalg RSA -
    keypass tomcat -storepass tomcat -keystore
    IntegrationSvr1.keystore

echo Exporting the Server's Certificate from Server's Keystore to an
    external file server.cer
%java_home%\bin\keytool -export -alias tomcat-sv -storepass tomcat -
    file server.cer -keystore IntegrationSvr1.keystore

echo Generating the Client Keystore in file client.keystore
%java_home%\bin\keytool -genkey -alias tomcat-cl -dname
    "CN=<servername>, OU=X, O=Y, L=Z, S=XY, C=YZ" -keyalg RSA -
    keypass tomcat -storepass tomcat -keystore client.keystore

:cleanup
```



# Troubleshooting

This chapter contains descriptions of potential solutions for problems you may encounter when using eGate Integrator within eInsight Enterprise Service Bus.

## What's in This Chapter

- [Error Messages](#) on page 225

---

## 10.1 Error Messages

### 10.1.1 OutOfMemory

Out-of-memory errors occurring during Project design are most frequently overcome by increasing the *heap size* of the editor you are using, or of Enterprise Designer itself (see [Options Setup](#) on page 44). Try increasing the heap size in increments of 50 MB, starting with the OTD editor.

If you are using an XSD OTD having a file size in excess of 1 MB, and experience out-of-memory errors that cannot be overcome by increasing the heap size, you may need to increase the *permanent generation* memory size. To do so, you must edit the **runed.bat** file (found in *ICAN-root\edesigner\bin*), adding the following command-line parameters after the command **runidew** (near the end of the file):

```
-J-XX:PermSize=192m -J-XX:MaxNewSize=128m -J-XX:MaxPermSize=192m
```

# Glossary

**BPEL**

BPEL (Business Process Execution Language), also known as BPEL4WS (Business Process Execution Language for Web Services), is an XML-based language designed to enable task sharing for either a distributed or grid computing environment. It combines and replaces **WSDL** and Microsoft's XLANG specification.

**Collaboration**

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

**Collaboration Definition**

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

**Connection**

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

**Connectivity Map**

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

**Constant**

A static name-value pair that is visible across a **Project**.

**CRM**

Acronym for Customer Relations Management.

**Data Cleansing**

Data must be "cleansed" of errors in structure and content for accurate and effective use in a database or data management systems.

**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. Although the dictionary does not contain actual data, it does contain essential information for managing the database. The data dictionary is often hidden from end users.

**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**

Refers to establishing the relationship and data flow pattern between source and target objects, usually within the context of relational database management systems (RDBMSs).

**Data Transformation**

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

**DBCS**

Acronym for Double-Byte Character Set.

**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**

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

**DTD**

A Document Type Definition (DTD) specifies how an associated document, written either in the Standard Generalized Markup Language (SGML) or of the Extensible Markup Language (**XML**), is to be processed.

**Enterprise Designer**

The **Project** design tool within eInsight Enterprise Service Bus.

**Enterprise Service Bus (ESB)**

A category of software, incorporating native Web services support, that provides a low-end alternative to a comprehensive integration broker suite—offering limited functionality, but less complexity and lower cost.

**Environment**

A collection of physical resources and their configurations that are used to host eInsight Enterprise Service Bus **Project** components. An Environment contains Logical Hosts and external systems.

**ERM**

Acronym for Enterprise Resource Management.

### **ETL**

A three-phase (extract/transform/load) process used, for example, to generate and maintain a central **Metadata** repository.

- **Extract** is the process of reading data from a source database and obtaining 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 transformed data into the target 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 of an application external to the ICAN Suite.

### **External System**

A representation of a computer system hosting an application external to the ICAN Suite.

### **Extraction**

The process of reading data from a source database and obtaining the desired subset of data (see **ETL**).

### **HTML**

HTML (HyperText Markup Language) is the set of markup symbols or codes (tags) inserted in a file intended for display on a Web page. HTML describes the content of the Web page (primarily text and graphics) only in terms of how it is to be displayed and interacted with.

### **HTTP**

HTTP (HyperText Transfer Protocol) is the set of rules for transferring files—text, graphics, audio, video—on the World Wide Web.

### **ICAN Suite**

The SeeBeyond Integrated Composite Application Network Suite.

### **Impact Analyzer**

A module within Enterprise Designer that analyzes and predicts the impact a specified change would have on other components in the **Project**.

### **Integration Server**

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

### **JMS IQ Manager**

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

**Link**

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

**Linked Message Destination**

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

**Logical Host**

A Logical Host contains the eInsight Enterprise Service Bus run-time components, including integration servers and message servers, that are installed on a host hardware platform.

**Management Agent**

Uses J2EE technology to manage and monitor an eInsight Enterprise Service Bus 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**

Metadata describes the structure and format of a particular set of data. **ETL** tools are used to generate and maintain a central metadata repository.

**Non-normalized Data**

Data that has not been converted to normalized data (see next entry).

**Normalized Data**

Data that has been processed to remove redundant or incorrect data structure and organization, thereby creating a maintainable data set that can be cross-referenced. Normalized data is not only easier to analyze but also easier to expand.

**Object Type Definition (OTD)**

Object Type Definitions contain the data structure and rules that define an object. OTDs are used in **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 eInsight Enterprise Service Bus system in SeeBeyond's **Enterprise Designer**.

**Query**

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

- **Choose** – With this simplified method, the database system presents a list of parameters from which you can choose. This method is the least flexible of the three 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 ability to make requests for information in the form of a stylized query using a query language. This is the most complex and powerful of the three methods.

### 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, although 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

Unprocessed data, as obtained from the source (see also **Non-normalized Data**).

### Relational Database (RDBMS)

Short for Relational Database Management System, most often referred to as RDBMS, in which data is stored in related tables and can be viewed in many different ways. Relational databases differ from flat-file databases, in which each database is self-contained as a single file or table.

### Repository

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

### SBCS

Acronym for Single-Byte Character Set.

### Service

Contains the information about executing a set of business rules. These business rules can be defined in a Java or 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.

### SI/SO

Acronym for Shift-In/Shift-Out.

### SOAP

SOAP (Simple Object Access Protocol) enables a program running in one operating system to communicate with another program running in either the same or a different operating system, using **HTTP** and **XML** as the mechanisms for information exchange.

**Subproject**

An independent **Project** that is included as part of another Project, and is displayed in the Enterprise Explorer tree as a “branch” beneath the main Project.

**Table**

Refers to data arranged in rows and columns, as in a spreadsheet. In **Relational Database (RDBMS)** 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 one or more subscribers. When the SeeBeyond **JMS IQ Manager** publishes a message on a topic, it ensures that all subscribers receive the message.

**Transformation**

The process of converting the data extracted from its source into the form required by its target program or system. This process includes **Data Cleansing**, **Data Mapping**, data normalization (see **Normalized Data**), and other sub-processes.

**UDDI**

UDDI (Universal Description, Discovery, and Integration) is an **WSDL**-based registry that enables businesses to list themselves and their services on the Internet.

**Version Control**

Features that maintain the integrity of a program or **Project** by controlling the ability of an individual to modify the program or Project, and providing an audit trail for accepted modifications.

**WSDL**

WSDL (Web Services Flow Language) is an **XML**-based language, derived from **SOAP**, used to describe the services a business offers via the Internet. WSDL provides the means of expressing business services in the **UDDI** registry.

**XML**

XML (Extensible Markup Language) is a superset of **HTML**, which also describes the content in terms of what data is being described. XML is *extensible* because—unlike HTML—the markup symbols are unlimited and self-defining.

**XSD**

XSD (XML Schema Definition) specifies how to formally describe the elements in an **XML** document. It is more powerful than, and generally replaces, the older Document Type Definition (DTD). eInsight Enterprise Service Bus makes use of **Object Type Definition (OTD)**s described in XSD, as well as DTD.

**XSLT**

XSLT (Extensible Stylesheet Language Transformation) is a language for transforming XML documents into other XML documents. It is designed for use as part of XSL, which is a stylesheet language for XML. eGate Integrator makes use of **Collaboration Definitions** coded in XSLT.

# Index

## A

ACL properties 77, 79–80, 83, 128–131  
 activation  
   Deployment Profile 182  
 automap command 175  
 automap feature 179

## B

BEA WebLogic 185  
 BPEL  
   definition 226

## C

certificate  
   defined 214  
 Certificate Authority (CA) 214  
 check in 66  
   without revisions 67  
 check out 68–69  
 Collaboration  
   definition 226  
   derived 227  
 Collaboration Definition  
   definition 226  
 component version  
   checked in 64  
   checked out 64  
   retrieved 64  
 connection 226  
 Connectivity Map  
   definition 226  
   Editor 86  
 constants 226  
   Environmental 132  
 conventions  
   writing 20  
 CRM 226  
 customizer 93

## D

data

cleansing 226  
 dictionary 226  
 integrity 227  
 mapping 227  
 non-normalized 229  
 normalized 229  
 raw 230  
   transformation 227  
 database, relational 230  
 deactivation  
   Deployment Profile 182  
 deployment  
   automapping 175, 179  
 Deployment Editor 175  
 Deployment Profile 174  
   activation 182  
   creation 176  
   deactivation 182  
   definition 227  
   map variables 180  
 derived Collaboration 227  
 DTD  
   definition 227  
   Wizard 107

## E

Editor  
   Connectivity Map 86  
   Deployment 175  
   OTD 100  
 Enterprise Designer  
   definition 227  
   enterprise explorer 47  
   menu bar 42  
   starting 40  
 Enterprise Explorer  
   Environments 48, 124  
   Projects 47, 74, 124  
 Enterprise Manager  
   Documentation 34  
   Interface 32  
   starting 31  
 Enterprise Monitor 35  
 Enterprise Service Bus (ESB)  
   definition 227  
 Environment 26  
   constants 132  
   definition 227  
 Environment Explorer 48, 124  
 EPR 227  
 error messages  
   OutOfMemory 225  
 ETL 228



## Index

eWay  
  definition 228  
external application 88  
  definition 228  
external system 26  
  definition 228  
extraction 228

## H

heap size 44  
HTTP  
  definition 228

## I

IBM WebSphere 188  
ICAN Suite 228  
Impact Analyzer  
  definition 228  
  usage 62  
Integration Server  
  definition 228  
Interfaces  
  Enterprise Manager 32

## J

JMS IQ Manager  
  definition 228

## K

keytool utility 215

## L

link 229  
load balancing  
  Web services 199  
Logical Host 26  
  definition 229

## M

Management Agent 229  
map variables 180  
menu bar 42  
message destination  
  definition 229  
  linked 229  
metadata 229  
Monitor

Enterprise 35

## O

Object Type Definition 229  
  wizard 108, 114, 119  
Object Type Definition (OTD) 97  
Open File command 103  
OTD 229  
  Editor 100  
  tester 102  
OTD Editor commands  
  Open File 103  
  Refresh OTD 103  
  Run Tester 103  
  Save as New Name 101  
  Save File 103  
  Tester 101  
  Toggle Reference Tab Panel 101  
OTD Wizard  
  DTD 107  
  WSDL 112  
  XSD 118  
OutOfMemory error 225

## P

Profile, Deployment 174  
Project  
  definition 229  
Project Explorer 47, 74, 124  
proxy server 38

## Q

query 229  
queue  
  definition 230

## R

rdbms 230  
Refresh OTD command 103  
relational database 230  
Repository 26  
  definition 230  
Run Tester command 103

## S

Save as New Name command 101  
Save File command 103  
Scheduler 89

## Index

- Screenshots 20
- security 139
- service
  - definition 230
- servlet context property 198
- SOAP 192
  - definition 230
- SSL
  - overview 214
- starting
  - Enterprise Designer 40
  - Enterprise Manager 31
- stateless session beans, deploying to IS 157
- subproject
  - definition 231
- supporting documents 21

## T

- table 231
- tester
  - OTD 102
- Tester command 101
- Toggle Reference Tab Panel command 101
- topic
  - definition 231
- transformation 231

## U

- UDDI 192
  - definition 231
  - registry 193
- undo check-out 67

## V

- variables
  - mapping 180
- version control 64
  - check in 64, 66
  - check out 64, 68–69
  - definition 231
  - history 65
  - retrieve 64
  - undo check-out 67

## W

- WebLogic 185
- WebSphere 188
- Wizard
  - OTD (DTD) 107

- OTD (WSDL) 112
- OTD (XSD) 118
- WSDL 191
  - definition 231
- Wizard 112

## X

- XML 191
  - definition 231
- XSD
  - definition 231
  - Wizard 118
- XSLT
  - definition 231