SeeBeyond ICAN Suite

ebXML Manager Composite Application User's Guide

Release 5.0.4

SEEBEYOND®

Version 20040604105941.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces you to this guide, its general purpose and scope, and its organization. It also provides sources of related documentation and information.

In this chapter

1.1 Overview

This user's guide provides instructions and background information for all users of the ebXML Manager Composite Application. This guide is designed for managers, system administrators, and others who use ebXML Protocol Manager.

The purpose of this guide is to help you do the following:

- Understand the nature of ebXML Protocol Manager.
- Understand the function of ebXML Protocol Manager.
- Understand the relationship of ebXML Protocol Manager to other components of the SeeBeyond Integrated Composite Application Network (ICAN) Suite.
- Learn about the ebXML Protocol Manager components and editors and how to use them in your environment.

1.2 Contents of This Guide

This guide is arranged as follows:

- **Chapter 1, "Introduction"** provides an overview of this document's purpose, contents, writing conventions, and supported documents.
- **Chapter 2, "Overview"** discusses general features and architecture of eXchange.
- **Chapter 2, "Installing ebXML Protocol Manager"** provides step-by-step instructions for installing the eXchange product and setting it up for use.
- **Chapter 3, "Implementation Scenario: ebXML"** provides step-by-step procedures for creating implementation scenarios that provide a sample of how eXchange can be used to achieve B2B solutions.

1.3 Writing Conventions

The following writing conventions are observed throughout this document.

Table 1 Writing Conventions

Text	Convention	Example
Names of buttons, files, icons, parameters, variables, methods, menus, and objects	**Bold** text	- Click **OK** to save and close. - From the **File** menu, select **Exit**. - Select the **logicalhost.exe** file. - Enter the **timeout** value. - Use the **getClassName()** method. - Configure the **Inbound** File eWay.
Command-line arguments, code samples	`Fixed` font. Variables are shown in ***bold italic***.	`bootstrap -p` ***password***
Hypertext links	**Blue** text	For more information, see **"Writing Conventions" on page 8**.

Additional Conventions

Windows Systems

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

Path Name Separator

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

1.4 Supporting Documents

For more information about eXchange and the ICAN Suite, refer to the following:

Title	Filename
SeeBeyond ICAN Site Installation Guide	ICAN_Install_Guide.pdf
SeeBeyond ICAN Suite Primer	Primer.pdf
SeeBeyond ICAN Suite Deployment Guide	Deployment_Guide.pdf
eGate Integrator Release Notes	eGate_Release_Notes.pdf
eGate Integrator Tutorial	eGate_Tutorial.pdf
eGate Integrator User's Guide	eGate_User_Guide.pdf
eGate Integrator System Administration Guide	Sys_Admin_Guide.pdf
eGate Integrator JMS Reference Guide	eGate_JMS_Reference.pdf
Oracle eWay Intelligent Adapter User's Guide	Oracle_eWay.pdf
HTTP(S) eWay Intelligent Adapter User's Guide	HTTPS_eWay.pdf
eGate Integrator User's Guide	eXchange_User_Guide.pdf
AS2 Manager Composite Application User's Guide	AS2_ProtocolManager_Guide.pdf
X12 OTD Library User's Guide	X12_OTD_Library.pdf
HIPAA OTD Library User's Guide	HIPAA_OTD_Library.pdf
UN/EDIFACT OTD Library User's Guide	EDIFACT_OTD_Library.pdf
Readme for ICAN 5.0.4	Readme.txt

1.5 Online Documents

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

http://www.adobe.com

1.6 The SeeBeyond Web Site

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

http://www.seebeyond.com

Chapter 2

Installing ebXML Protocol Manager

This chapter provides the prerequisites and steps for installing ebXML Manager Composite Application.

2.1 System Requirements

This section lists system requirements and database requirements. The *SeeBeyond ICAN Suite Installation Guide* and the **Readme.txt** file, available on the product media and via Enterprise Manager (Documentation tab), contain up-to-date operating system requirements for each supported platform.

2.1.1. Platform Support

eXchange supports the following operating systems:

- Microsoft Windows Server 2003, Windows XP SP1a, and Windows 2000 SP3 or SP4
- Sun Solaris 8 and 9, with required patches
- HP Tru64 V5.1A with required patches
- HP-UX 11.0 and 11i (PA-RISC), with required patches and parameter changes
- IBM AIX 5.1L and 5.2 (either 64-bit kernel or 32-bit kernel with 64-bit extension), with required maintenance level patches
- Red Hat Linux 8 (Intel *x*86) and Linux Advanced Server 2.1 (Intel *x*86)

2.1.2. Database Support

Database for eXchange Partner Management and Message Tracking

The eXchange database provides a run-time persistent store for trading partner management and message tracking. eXchange supports the following databases:

- Oracle 8.1.7
- Oracle 9.01
- Oracle 9.2

2.2 Installation Steps

The steps for installing ebXML Protocol Manager are the same as for other products in the ICAN Suite. You can find general product installation instructions in the *ICAN Suite Installation Guide*, which is available on the product media and can also be accessed via Enterprise Manager (Documentation tab).

2.2.1. Uploading ebXML Protocol Manager to the Repository

Before you begin

- A Repository server must be running on the machine where you will be uploading the product files.
- The following ICAN **.sar** files must have already been uploaded to this Repository:
  - eGate Enterprise Designer (**eGate.sar**) 5.0.4
  - Batch eWay adapter (**BatcheWay.sar**) 5.0.4
  - Oracle eWay adapter (**OracleeWay.sar**) 5.0.4
  - File eWay adapter (**FileeWay.sar**) — This not an installation requirement, but it is required by the sample implementation
  - eXchange Integrator (**eXchange.sar**) 5.0.4
  - HTTP(S) eWay adapter (**HTTPeWay.sar**) 5.0.4
  - Secure Message Extension (**SMEWebServices.sar**) 5.0.4

Note: *SMEWebServices.sar is required for such features as encryption/decryption, signature verification, certificate authentication, and compression/decompression.*

To upload ebXML Protocol Manager product files to the Repository

1. On a Windows machine, start a Web browser and point it at the machine and port where the Repository server is running:

   ```
   http://<hostname>:<port>
   ```

   where

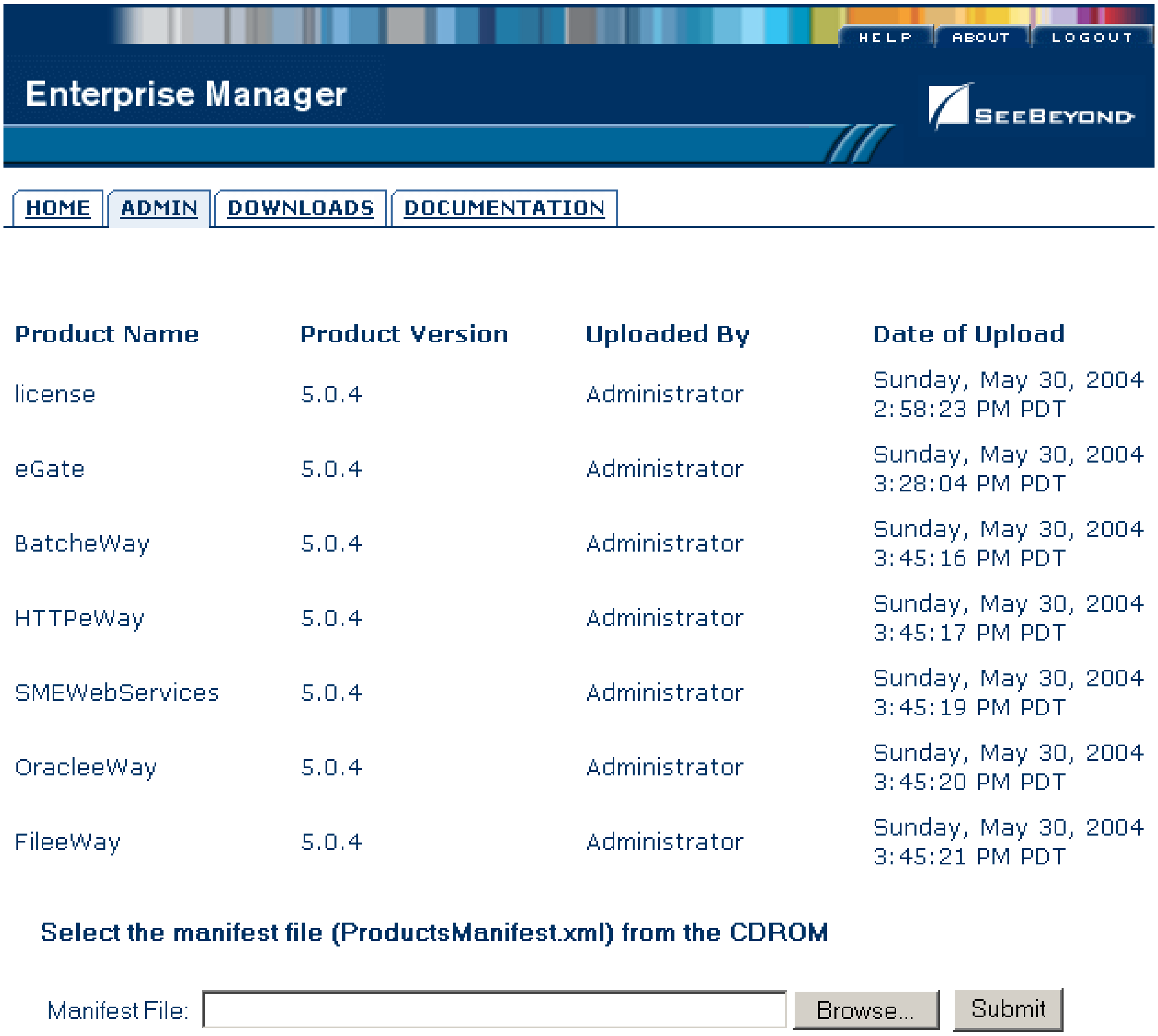
   - *<hostname>* is the name of the machine running the Repository server.
   - *<port>* is the starting port number assigned when the Repository was installed.

   For example, the URL you enter might look like either of the following:

   ```
   http://localhost:12001
   http://serv1234.company.com:19876
   ```

2. On the Enterprise Manager **SeeBeyond Customer Login** page, enter your username and password.
3. When Enterprise Manager responds, click the **ADMIN** tab. See Figure 1.

Figure 1 Enterprise Manager ADMIN page



Product Name	Product Version	Uploaded By	Date of Upload
license	5.0.4	Administrator	Sunday, May 30, 2004 2:58:23 PM PDT
eGate	5.0.4	Administrator	Sunday, May 30, 2004 3:28:04 PM PDT
BatcheWay	5.0.4	Administrator	Sunday, May 30, 2004 3:45:16 PM PDT
HTTPeWay	5.0.4	Administrator	Sunday, May 30, 2004 3:45:17 PM PDT
SMEWebServices	5.0.4	Administrator	Sunday, May 30, 2004 3:45:19 PM PDT
OracleeWay	5.0.4	Administrator	Sunday, May 30, 2004 3:45:20 PM PDT
FileeWay	5.0.4	Administrator	Sunday, May 30, 2004 3:45:21 PM PDT

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

Manifest File:

4 In the ADMIN page, click **Browse**.

5 In the **Choose file** dialog, click **ProductsManifest.xml**, and then click **Open**.

6 In the ADMIN page, click **Submit**.

The lower half of the ADMIN page lists the product files you are licensed to upload.

7 In the Products column, find **ebXML Manager Composite Application**, and then click the **Browse** button for it.

8 In the **Choose file** dialog, click **ebXML_Protocol_Manager.sar** and then click **Open**.

9 Repeat the previous two steps for other eXchange-related product **.sar** files you are licensed to upload, such as OTD libraries for ASC/X12 or UN/EDIFACT.

10 For documentation and samples, also upload the corresponding [...]**Docs.sar** files, such as **ebXML_Protocol_ManagerDocs.sar** (and X12_OTD_Docs.sar, and so forth).

11 In the ADMIN page, click the |upload now ⁚·| button.

2.2.2. Refreshing Enterprise Designer with ebXML Protocol Manager

Before you begin

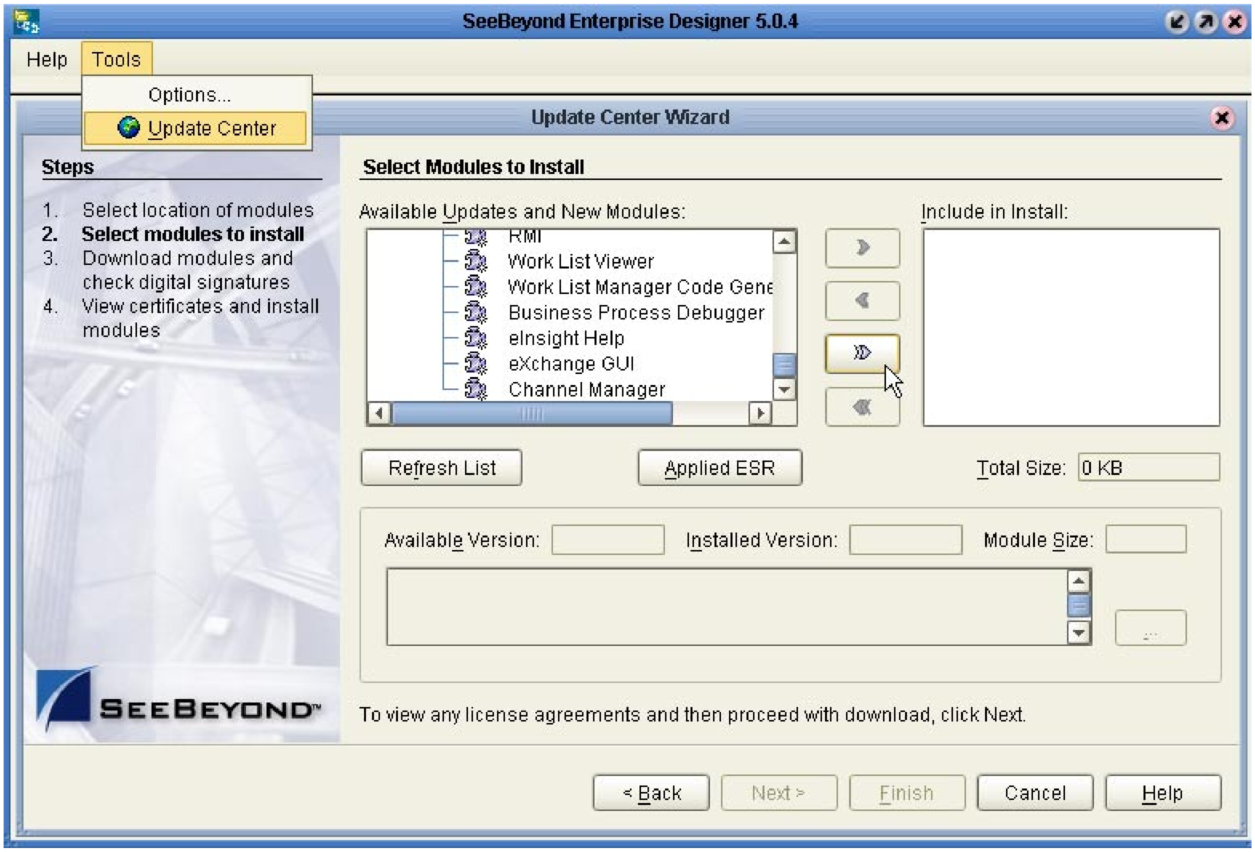
- You must have already downloaded and installed Enterprise Designer.
- A Repository server must be running on the machine where you uploaded the ebXML Protocol Manager product files.

To refresh an existing installation of Enterprise Designer

1. Start Enterprise Designer.
2. On the **Tools** menu, click **Update Center**.

   The Update Center shows a list of components ready for updating. See Figure 2.

Figure 2 Update Center Wizard: Select Modules to Install



SeeBeyond Enterprise Designer 5.0.4

Update Center Wizard

Steps

1. Select location of modules
2. **Select modules to install**
3. Download modules and check digital signatures
4. View certificates and install modules

Select Modules to Install

Available Updates and New Modules:

- RMI
- Work List Viewer
- Work List Manager Code Gene
- Business Process Debugger
- eInsight Help
- eXchange GUI
- Channel Manager

Include in Install:

Total Size: 0 KB

Available Version:

Installed Version:

Module Size:

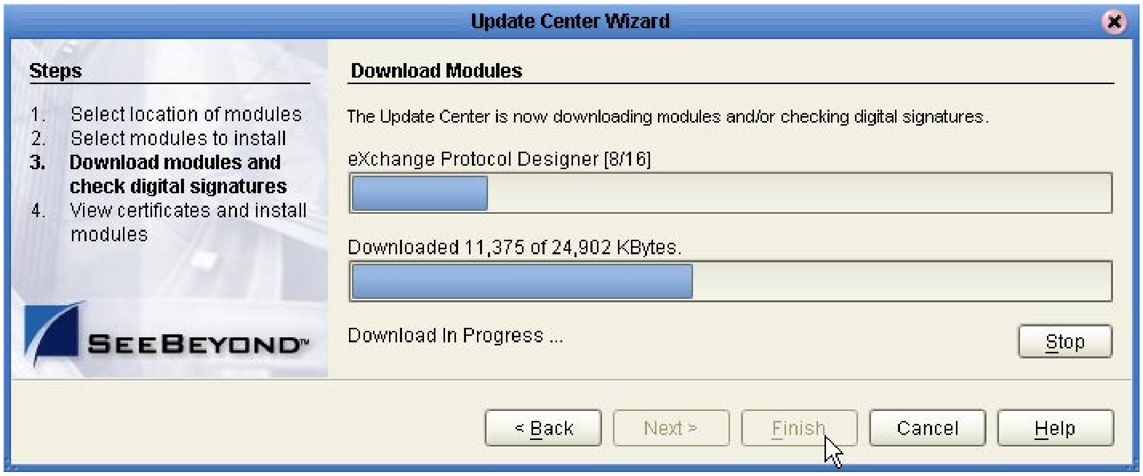
To view any license agreements and then proceed with download, click Next.

3. Click **Add All** (the button with a doubled chevron pointing to the right).

   All modules move from the Available/New pane to the **Include in Install** pane.
4. Click **Next** and, in the next window, click **Accept** to accept the license agreement.

   The wizard shows you the progress of the download. See Figure 3.

Figure 3 Update Center Wizard: Progress Bars



Update Center Wizard

Steps

1. Select location of modules
2. Select modules to install
3. **Download modules and check digital signatures**
4. View certificates and install modules

Download Modules

The Update Center is now downloading modules and/or checking digital signatures.

eXchange Protocol Designer [8/16]

Downloaded 11,375 of 24,902 KBytes.

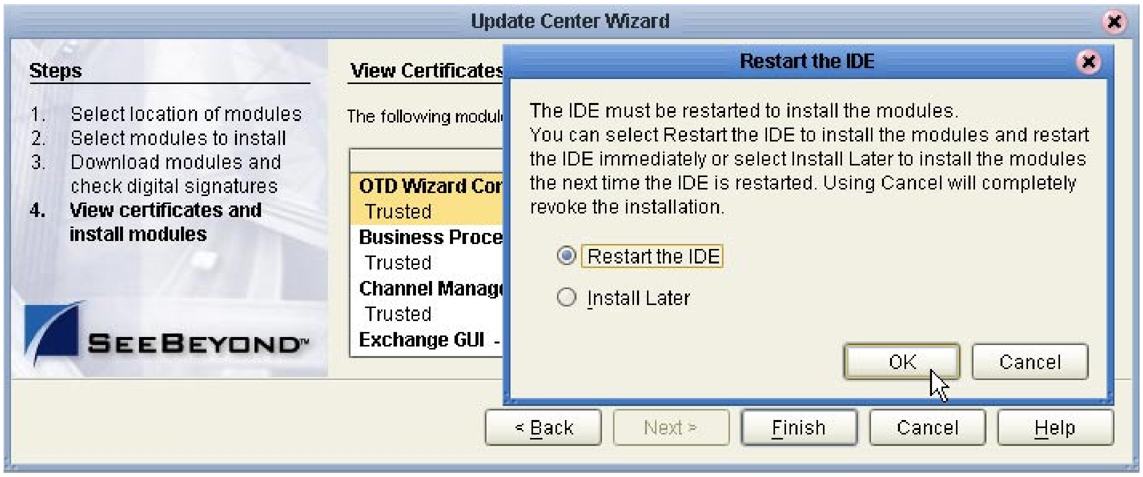
Download In Progress ...

5 When the progress bars indicate the download has ended, click **Next**.

6 Review the certificates and installed modules, and then click **Finish**.

7 When prompted to restart Enterprise Designer, click **OK**. See Figure 4.

Figure 4 Update Center Wizard: Restart Enterprise Designer



Update Center Wizard

Steps

1. Select location of modules
2. Select modules to install
3. Download modules and check digital signatures
4. **View certificates and install modules**

View Certificates

The following modul

OTD Wizard Cor
Trusted
Business Proce
Trusted
Channel Manag
Trusted
Exchange GUI -

Restart the IDE

The IDE must be restarted to install the modules.
You can select Restart the IDE to install the modules and restart the IDE immediately or select Install Later to install the modules the next time the IDE is restarted. Using Cancel will completely revoke the installation.

● Restart the IDE

○ Install Later

When Enterprise Designer restarts, the installation of ebXML Manager Composite Application is complete, and you can use all eXchange tools provided on the Enterprise Designer framework.

2.3 Database Scripts

The eXchange database schema collects and persists data about your trading partner profiles, and also allows you to track message delivery history. eXchange provides database scripts to create and upgrade these database schemas for eXchange. For information, see the *eXchange Integrator User's Guide*.

2.4 Additional Policy JAR Files Required to Run SME

For strong encryption/decryption and signatures/verification, using

libraries supplied with Secure Messaging Extension (SME), you must download and apply additional policy **.jar** files. The type of **.jar** files required depends on the JVM are using. Refer to your JVM vendor for exact details on the specific policy **.jar** file requirements.

Use Table 3 to determine which JRE is included in the eGate logical host.

Table 2 JRE Versions Listed by Operating System

Operating System	JRE	URL
Windows, Solaris, HP-UX, Linux, Tru64	1.4.2	**http://java.sun.com/j2se/1.4.2/download.html**
AIX	1.4.1	**http://java.sun.com/products/archive/j2se/1.4.1_07/index.html**

To download the required policy.jar files

1. Scroll to the bottom of the web page listed in Table 3 for your logical host's JRE.
2. Click the DOWNLOAD link for **Unlimited Strength Jurisdiction Policy Files 1.4.2**. (or, for AIX, **Unlimited Strength Jurisdiction Policy Files 1.4.1**).
3. Click the link to download the **.zip** file containing the required policy **.jar** files (in other words, for JRE 1.4.2, the link Download **jce_policy-1_4_2.zip**; or, for JRE 1.4.1, the link Download **jce_policy-1_4_1.zip**)
4. Extract the following required policy **.jar** files:
   - **local_policy.jar**
   - **US_export_policy.jar**
5. Then, for each of your logical hosts, replace the versions of these files in:

   `<logicalhost>\jre\lib\security\`
6. In addition, if you are running a repository on AIX, also replace the versions of these files in:

   `<AIXrepository>/jre/1.4.x/security/`

For complete information on SME, see the *Secure Messaging Extension User's Guide*.

libraries supplied with Secure Messaging Extension (SME), you must download and apply additional policy **.jar** files. The type of **.jar** files required depends on the JVM are using. Refer to your JVM vendor for exact details on the specific policy **.jar** file requirements.

Use Table 3 to determine which JRE is included in the eGate logical host.

Table 3 JRE Versions Listed by Operating System

Operating System	JRE	URL
Windows, Solaris, HP-UX, Linux, Tru64	1.4.2	**http://java.sun.com/j2se/1.4.2/download.html**
AIX	1.4.1	**http://java.sun.com/products/archive/j2se/1.4.1_07/index.html**

To download the required policy.jar files

1. Scroll to the bottom of the web page listed in Table 3 for your logical host's JRE.
2. Click the DOWNLOAD link for **Unlimited Strength Jurisdiction Policy Files 1.4.2**. (or, for AIX, **Unlimited Strength Jurisdiction Policy Files 1.4.1**).
3. Click the link to download the **.zip** file containing the required policy **.jar** files (in other words, for JRE 1.4.2, the link Download **jce_policy-1_4_2.zip**; or, for JRE 1.4.1, the link Download **jce_policy-1_4_1.zip**)
4. Extract the following required policy **.jar** files:
   - **local_policy.jar**
   - **US_export_policy.jar**
5. Then, for each of your logical hosts, replace the versions of these files in:

   ```
   <logicalhost>\jre\lib\security\
   ```
6. In addition, if you are running a repository on AIX, also replace the versions of these files in:

   ```
   <AIXrepository>/jre/1.4.x/security/
   ```

For complete information on SME, see the *Secure Messaging Extension User's Guide*.

Additional JAR Files for ebXML Encryption

For ebXML, you must also download and apply an additional **.jar** file, xss4j.jar, for XML security. This can be downloaded, as part of the XML Security Suite, from the following Web site:

http://www.alphaworks.ibm.com/tech/xmlsecuritysuite

The only file needed is **xss4j.jar**. After extracting it, copy it to the following location for each logical host:

```
<logicalhost>\stcis\lib\
```

Chapter 3

Implementation Scenario: ebXML

This chapter provides an implementation scenario showing how eXchange can be used to achieve B2B solutions using the ebXML protocol.

The steps for the sample implementation occur in these phases:

Initial Setup Steps	In these steps, you ensure that prerequisites are met, obtain the necessary sample materials, extract sample files, and import the sample projects. See section 3.1 **on page 18**.
Design Steps in Enterprise Designer	In these steps, you use Enterprise Designer to add and configure externals, view components in the B2B host project, and it, creating an eXchange Service for the B2B host. Then you view the components in the main project and activate it. See section 3.2 **on page 19**.
Design Steps in ePM	In these steps, you use the eXchange Partner Management (ePM) facility to create a trading partner, view and specify parameter values for it, and activate it. See section 3.3 **on page 28**.

Overview of the Sample ebXML Implementation

The sample implementation combines inbound and outbound message processing. It makes use of the following ebXML protocol processes supplied by SeeBeyond:

Table 4 B2B Protocol Processes Used in the ebXML Sample Implementation

For Inbound Messages	For Outbound Messages	Other
ebXML_Inbound	DcpFinderCollab	
ebXmlIdentification	ToInternalDeliverySelector	ToInternalJMSDeliveryProtocol
ValidateCollab		ToInternalFileDeliveryProtocol
DecryptCollab	EncryptCollab	SendExternalCollab
VerifyCollab	SignCollab	
SignAckCollab	VerifyAckCollab	
ExtractPayloadCollab	PackCollab	
	RetrieveAckCollab	
TrackerCollab		JMSErrorHandler
PackMshSignalCollab		ErrorHandlerSelector
TrackAckCollab		OutboundHTTPClient
Base64EncodeCollab		

3.1 Initial Setup Steps

Initial setup steps:

- **Installing the Sample Files for ebXML on page 18**
- **Setting Up the Sample Environment for Berlin (Company B) on page 19**
- **Importing the Sample Projects on page 19**

3.1.1 Installing the Sample Files for ebXML

These steps assume the existence of a temporary eXchange directory for sample files, such as **C:\temp\eXchange\**. You will extract the sample files to this directory so that you can conveniently access the files in later procedures.

To install the sample files

Before you begin: Your repository must already be running, and you must be logged in to Enterprise Manager. If you have already uploaded the documentation for eXchange, you can skip steps 1 and 2 and start with step 3.

1. In the ADMIN tab, if you have not already done so, browse to the [...]\Documentation**ProductsManifest.xml** file and submit it.
2. In the ADMIN tab, if you have not previously done so, browse to the **ebXML_Protocol_ManagerDocs.sar** file, select it, and click the **upload now** button.
3. In the DOCUMENTATION tab, in the Products window, click **ebXML Manager Composite Application**.
4. In the window that appears on the right side, click **Download Sample**.
5. Preserving file paths, extract the files to your **C:\temp\eXchange** directory.

Result: The following directories and files are created:

```
C:\temp\eXchange\Sample\Projects\ebXML_All.zip
C:\temp\eXchange\Sample\Projects\FileToJmsFeeder.zip
C:\temp\eXchange\Sample\Projects\JmsErrorToFileReceiver.zip
C:\temp\eXchange\Sample\Projects\JmsPayloadToInternalReceiver.zip
C:\temp\eXchange\Sample\Projects\Environments.zip
C:\temp\eXchange\Sample\TradingPartners\ebXML_AtlantaHost_TP.xml
C:\temp\eXchange\Sample\TradingPartners\ebXML_BerlinHost_TP.xml
C:\temp\eXchange\Sample\Data\ebXML_Atlanta_1-XML.~in
C:\temp\eXchange\Sample\Data\ebXML_Atlanta_2-XML-binary.~in
C:\temp\eXchange\Sample\Data\ebXML_Berlin_1-XML.~in
C:\temp\eXchange\Sample\Data\ebXML_Berlin_2-XML-binary.~in
C:\temp\eXchange\Sample\Data\payload.binary
C:\temp\eXchange\Sample\Data\payload.xml
C:\temp\eXchange\Sample\Data\Out\
```

3.1.2 Importing the Sample Projects

To import the sample projects

Before you begin: Your repository must already be running, and you must be logged in to Enterprise Designer. If your repository already has a project at the root level whose name is identical to one of the seven projects you will be importing, you must delete or rename it before you start.

1. In Project Explorer, right-click the repository and, on the popup menu, click: **Import**
2. In the **Import Manager** dialog, browse to the folder where you installed the sample files (such as `C:\temp\eXchange\Sample\Projects`), select **ebXML_All.zip**, and click **Open**.
3. Select all four projects, click **Import**, and then click **OK** to clear the confirmation.
4. Repeat the previous two steps to import the projects in: **FileToJMSFeeder.zip**; **JmsErrorToFileReceiver.zip**; and **JmsPayloadToInternalReceiver.zip**.
5. Close the Import Manager dialog.

Result: The Project Explorer tree now displays seven new projects: ebXML_AtlantaHost, ebXML_B2B Templates, ebXML_BerlinHost, ebXML_Protocol; and FileToJMSFeeder, JmsErrorToFileReceiver, and JmsPayloadToInternalReceiver.

3.2 Design Steps in Enterprise Designer

For the ebXML sample implementation, design-time steps in Enterprise Designer consist of the following:

- **Setting Up the Sample Environment for Berlin (Company B) on page 19**
- **Viewing and Activating the B2B Host Project for Berlin on page 21**
- **Configuring the eXchange Service with Crypto Information on page 23**
- **Creating and Activating the Project Deployment Profiles on page 24**

3.2.1 Setting Up the Sample Environment for Berlin (Company B)

The sample assumes you will use default configurations for all servers where possible, and that you will make any changes where needed. For example:

- You must create a new outbound Oracle external and configure it, even if you imported the sample environment; the sample parameters are for reference only.
- If you use anything other than a SeeBeyond Integration Server on ports 18000–18009, make adjustments in steps 3 (ports) and/or 4 (type of Integration Server).
- If you want your HTTP client to use SSL, see the *HTTP(S) eWay Intelligent Adapter User's Guide* for eWay settings and Integration Server Web Server configuration.

To create the sample environment

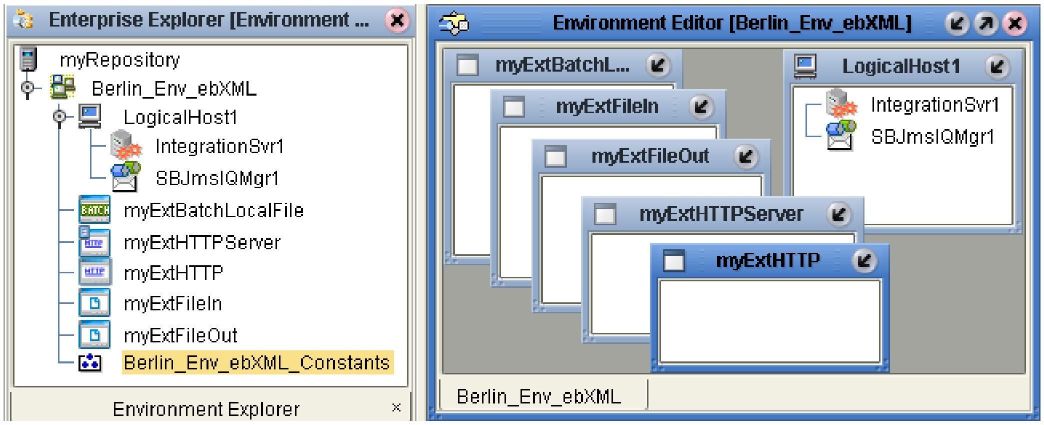
1. In Enterprise Designer, near the lower left of the window, click the **Environment Explorer** tab.
2. In the Environment Explorer tree, right-click the repository and, on the popup context menu, click **New Environment**
   - Rename the newly created environment to **Berlin_Env_ebXML**.
3. Right-click Berlin_Env_ebXML and, on the menu, click: **New Logical Host**
   - Retain the default name: LogicalHost1

Tip: *For a second or subsequent logical host: Right-click it and open its properties, click **Logical Host Configuration** and change the value **Logical Host Base Port** to a larger multiple of 1000 (19000 if ports 19000-19009 are unused; otherwise 20000, or 21000), and then close the properties sheet.*

4. Right-click LogicalHost11 and click: **New SeeBeyond Integration Server**
   - Retain the default name: IntegrationSvr1
5. Right-click LogicalHost1 and click: **New SeeBeyond JMS IQ Manager**
   - Retain the default name: SBJmsIQMgr1
6. Right-click Berlin_Env_ebXML > **New BatchLocalFile External System**
   - Name it **myExtBatchLocalFile** and click OK.
7. Right-click Berlin_Env_ebXML > **New File External System**
   - Name it **myExtFileIn**, set it to **Inbound File eWay**, and click OK.
8. Right-click Berlin_Env_ebXML > **New File External System**
   - Name it **myExtFileOut**, set it to **Outbound File eWay**, and click OK.
9. Right-click Berlin_Env_ebXML > **New HTTP Server External System**
   - Name it **myExtHTTPServer** and click OK.
10. Right-click Berlin_Env_ebXML > **New HTTP External System**
    - Name it **myExtHTTP** and click OK.
11. Right-click Berlin_Env_ebXML > **New Constant**
    - In the **Create a Constan**t dialog, name it **DataRootDirectory**, give it the value **C:\temp\eXchange\Sample\Data\Berlin** (if you installed the samples to a location other than C:\temp\eXchange, substitute appropriately), and click **OK**; then close the **Variables and Constants** dialog box.
12. On the main toolbar, click Save All.

Result: The environment, named **Berlin_Env_ebXML**, now has all but two of the externals needed by the projects. Steps for the outbound Oracle external are provided in the following procedure, and the final external will be created by activating the project containing the B2B host.

Figure 5 Sample Environment, Before Configuration



To create and configure the Oracle external

Before you begin: Your eXchange 5.0.4 Oracle database must be accessible, and you must you know its SID, username, and password.

1. In the Environment Explorer tree, right-click Berlin_Env_ebXML and, on the popup context menu, click **New Oracle External System**
   - Name it **myExtOracleOut**, designate it **Outbound Oracle eWay**, and click OK.
2. Right-click myExtOracleOut and configure properties appropriately. For example:
   - **DatabaseName:** *exch50 (change this to the SID for your eXchange Oracle database)*
   - DataSourceName: **local**
   - **Password:** *(replace this with the password for your eXchange 5.0.4 database user)*
   - **PortNumber: 1521** *(change this only if your Oracle administrator changed the default)*
   - **ServerName:** *myMachine (change this to the hostname of the Oracle server machine)*
   - **User:** *ex_admin (change this to the username for your eXchange database user)*
3. When all properties have been configured correctly for your site, click OK.
4. Collapse the Berlin_Env_ebXML tree, click Save All, and close all canvases.

Result: **Berlin_Env_ebXML** now has all but one of the externals needed by the projects.

The final external, an eXchange service, will be created by activating the project that contains the B2B host.

3.2.2 Viewing and Activating the B2B Host Project for Berlin

In the Project Explorer tree, open the B2B host project (named **ebXML_BerlinHost**) to display its four components. Below is a summary of the B2B host's contents. Activating this project will create an eXchange service that acts as a channel manager and provides a connection to the message tracking application and the eXchange database.

Components of the B2B host project

- **myHost_MsgingService** is the only message service used by the B2B host. It contains eight messaging actions, alternating inbound and outbound.
- **BerlinHost** is the B2B host itself:
  - Its Business Protocols references two services (both under the ebXML MAD): **myHost_MsgingService** (from the current project), and **urn:oasis:names:tc:ebxml-msg:service**
  - It defines only one external delivery channel (for the ebXML MAD), named **ebxml_delivery_channel1**. For transport to and from trading partners, this channel references the standard SeeBeyond-supplied HTTP transport attributes definitions (TADs).
  - It defines only one internal delivery channel (IDC): **Sender_IDC**. This uses JMS to communicate in the Sender (toInternal) direction.
- **BerlinHost_CMap** is the map whose activation will create the eXchange service.
  - Its only input is an instance of BerlinHost, with two outbound connections.
  - Its only output is an instance of Oracle, with two inbound connections.
  - Connecting to both is an instance of a SeeBeyond-supplied tracking application.

To activate the B2B host, creating the eXchange service

Before you begin: Your environment must contain a well-configured Oracle external (see the preceding procedure), and the environment must be named Berlin_Env_ebXML— that is, it must correspond to the name of your host project.

1. Right-click **ebXML_BerlinHost** and, on the popup context menu, point at **New** and click **Deployment Profile**
2. Keep the default name (Deployment1), point it at Berlin_Env_ebXML, and click OK.

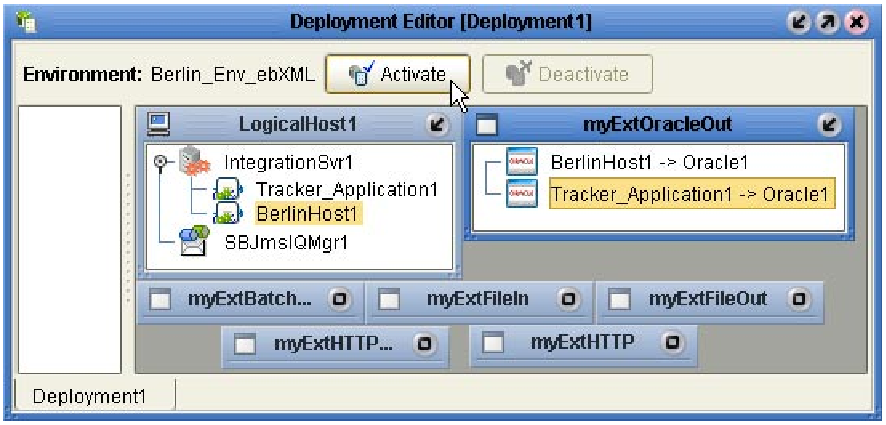
   The Deployment Editor opens. Its left pane has two services and two Oracle eWays.
3. On the right side, minimize all windows except LogicalHost1 and myExtOracleOut.
4. One by one, drag the two services into LogicalHost1 and under **IntegrationSvr1**.
5. One by one, drag the two Oracle eWays into **myExtOracleOut**. See Figure 6.

Tip: *If myExtOracleOut refuses to accept eWays, it may be an indication of:*

- *The Oracle database instance it references is inaccessible. Ensure it is running and that the myExtOracleOut properties match its hostname, SID, username, and password. If necessary, see* **"To create and configure the Oracle external" on page 21**.

- *It was misdefined as inbound. Delete myExtOracleOut and re-create it as outbound. Then: Click* ***Save All****, followed by* ***Refresh All from Repository****.*

Figure 6 Deployment Profile for B2B Host, Before Activation



6 Click Save All, and then click **Activate**.

7 In response to the dialog box, click **No**; that is, do *not* apply to Logical Host(s).

Result: A new external is created, named **BerlinHost1 eXchange Service**. The projects now have all the externals they need. Save all of your work, close all canvases, and click **Refresh All from Repository**.

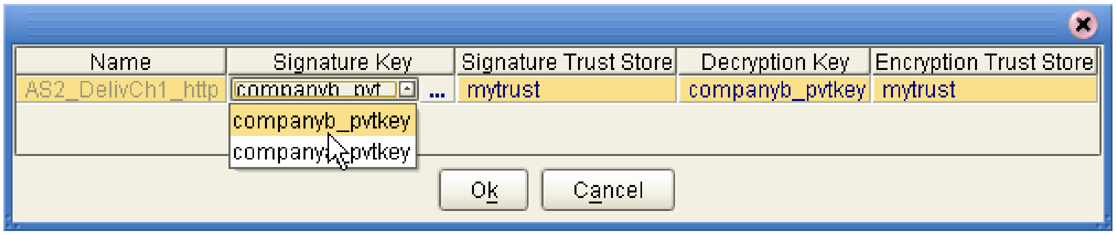
3.2.3 Configuring the eXchange Service with Crypto Information

Since the sample assumes you will be using cryptographic features (encryption, decryption, signatures and verifications), additional steps are required for configuring the eXchange service.

To associate the eXchange service with the private key for "Company B"

1 In the Environment Explorer tree, right-click **BerlinHost1 eXchange Service** and, on the popup context menu, click **Properties** to configure the public and private keys for the B2B host's delivery channel (**ebxml_delivery_channel1**). See Figure 7.

Figure 7 Configuring Private Keys and Trust Stores



Name	Signature Key	Signature Trust Store	Decryption Key	Encryption Trust Store
AS2_DelivCh1_http	companyb_pvt	mytrust	companyb_pvtkey	mytrust

2 For Signature Key, click **companyb_pvtkey**

Tip: *If companyb_pvtkey does not appear in the drop-down list, click the ellipsis [...] and, in the Signature Key dialog, click* ***Import****. Using alias* ***companyb_pvtkey****, import*

*C:\temp\eXchange\Sample\Crypto\**CompanyB-Key.p12** with the following password (all-lowercase): **companyb***

3 For Signature Trust Store, click **mytrust**

Tip: *If mytrust does not appear in the drop-down list, click the ellipsis [...] and, in the Signature Trust Store dialog, click New, enter the alias **mystrust**, and click OK.*

4 For Decryption, click **companyb_pvtkey**

5 For Encryption Trust Store, click **mytrust**

6 Click **OK** to close the dialog box.

7 If a logical host were running, you would also need to do the following:
In the environment tree, right-click **LogicalHost1** and, on the menu, click **Apply**

8 Click **Save All**, and then close all canvases.

Result: Cryptographic information is now associated with the ebxml_delivery_channel1 delivery channel for this eXchange service.

3.2.4. Creating and Activating the Project Deployment Profiles

To activate the main ebXML project

Before you begin: Your environment must be named Berlin_Env_ebXML (that is, it must correspond to the name of your host project), and it must contain an eXchange service; if necessary, see the **procedure on page 22**.

1 In the Project Explorer tree, right-click **ebXML_Protocol** and, on the popup context menu, point at New and click **Deployment Profile**

2 Keep the default name (Deployment1), point it at Berlin_Env_ebXML, and click OK.

The Deployment Editor’s left pane displays: A long list of services; a topic (JMSErrorHandler1->Host...) and two queues; an inbound eWay for HTTPServer; outbound eWays for HTTP and BatchLocalFile; and eWays for one inbound and four outbound eXchange services.

3 One by one, drag all services into LogicalHost1 and under **IntegrationSvr1**.

4 One by one, drag the topic and queues into LogicalHost1 and under **SbJmsIQMgr1**.

5 Drag the inbound HTTPServer eWay into myExtHTTPServer.

6 Drag the outbound HTTP eWay into myExtHTTP.

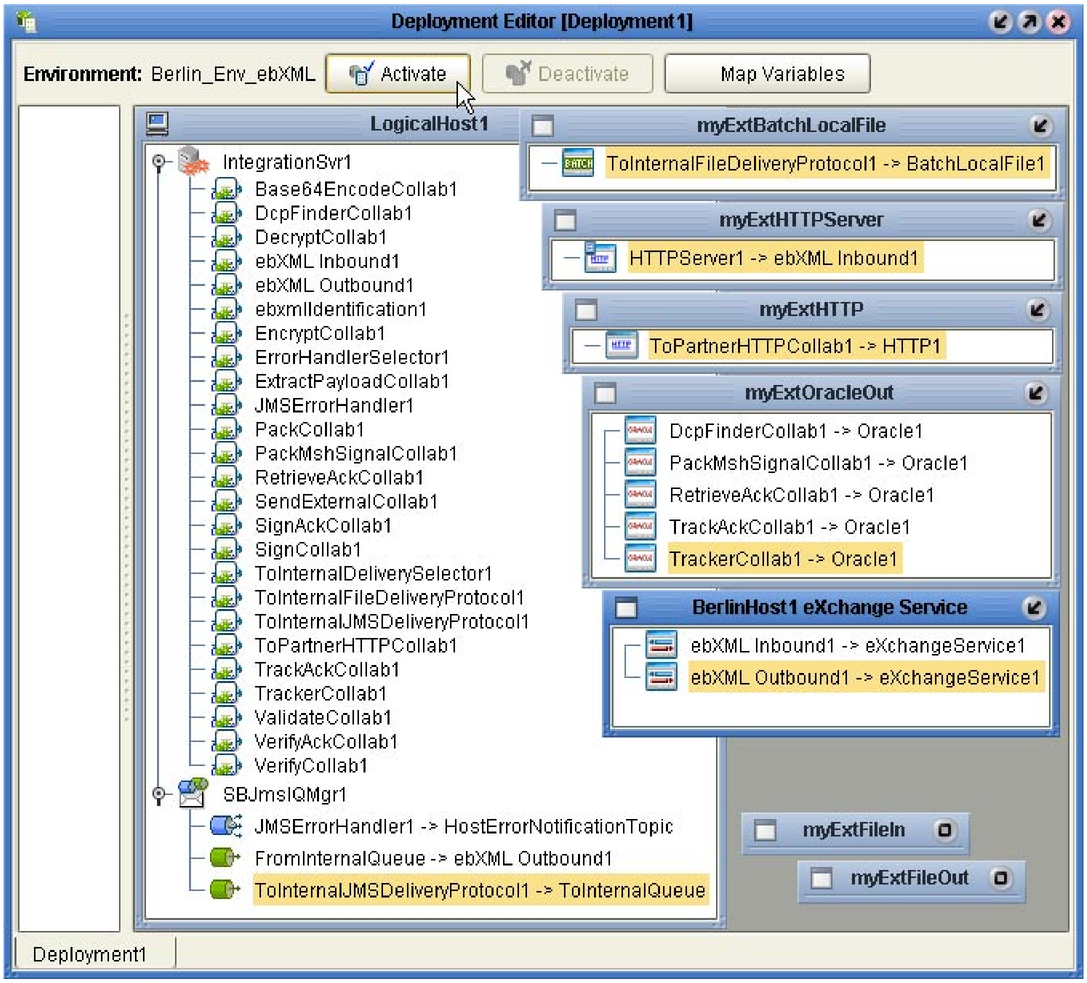
7 Drag the outbound BatchLocalFile eWay into myExtBatchLocalFile.

8 Drag the eXchange Service eWays into Berlin2Host1 Exchange Service. When you drag and drop the inbound service, specify **ebXML** as the protocol.

9 When Deployment1 is complete—that is, when all components in the ebXML_Protocol project are associated with corresponding servers—click **Activate**. See Figure 8.

Tip: *Successful activation will take some time. If activation fails because of an improper configuration of myExtHTTP, return to the environment, open the properties of the HTTP external, and provide it with dummy values for truststore and password.*

Figure 8 Deployment Profile for ebXML_Protocol (BerlinHost)



Environment: Berlin_Env_ebXML

LogicalHost1

- IntegrationSvr1
  - Base64EncodeCollab1
  - DcpFinderCollab1
  - DecryptCollab1
  - ebXML Inbound1
  - ebXML Outbound1
  - ebxmlIdentification1
  - EncryptCollab1
  - ErrorHandlerSelector1
  - ExtractPayloadCollab1
  - JMSErrorHandler1
  - PackCollab1
  - PackMshSignalCollab1
  - RetrieveAckCollab1
  - SendExternalCollab1
  - SignAckCollab1
  - SignCollab1
  - ToInternalDeliverySelector1
  - ToInternalFileDeliveryProtocol1
  - ToInternalJMSDeliveryProtocol1
  - ToPartnerHTTPCollab1
  - TrackAckCollab1
  - TrackerCollab1
  - ValidateCollab1
  - VerifyAckCollab1
  - VerifyCollab1
- SBJmsIQMgr1
  - JMSErrorHandler1 -> HostErrorNotificationTopic
  - FromInternalQueue -> ebXML Outbound1
  - ToInternalJMSDeliveryProtocol1 -> ToInternalQueue

myExtBatchLocalFile

- ToInternalFileDeliveryProtocol1 -> BatchLocalFile1

myExtHTTPServer

- HTTPServer1 -> ebXML Inbound1

myExtHTTP

- ToPartnerHTTPCollab1 -> HTTP1

myExtOracleOut

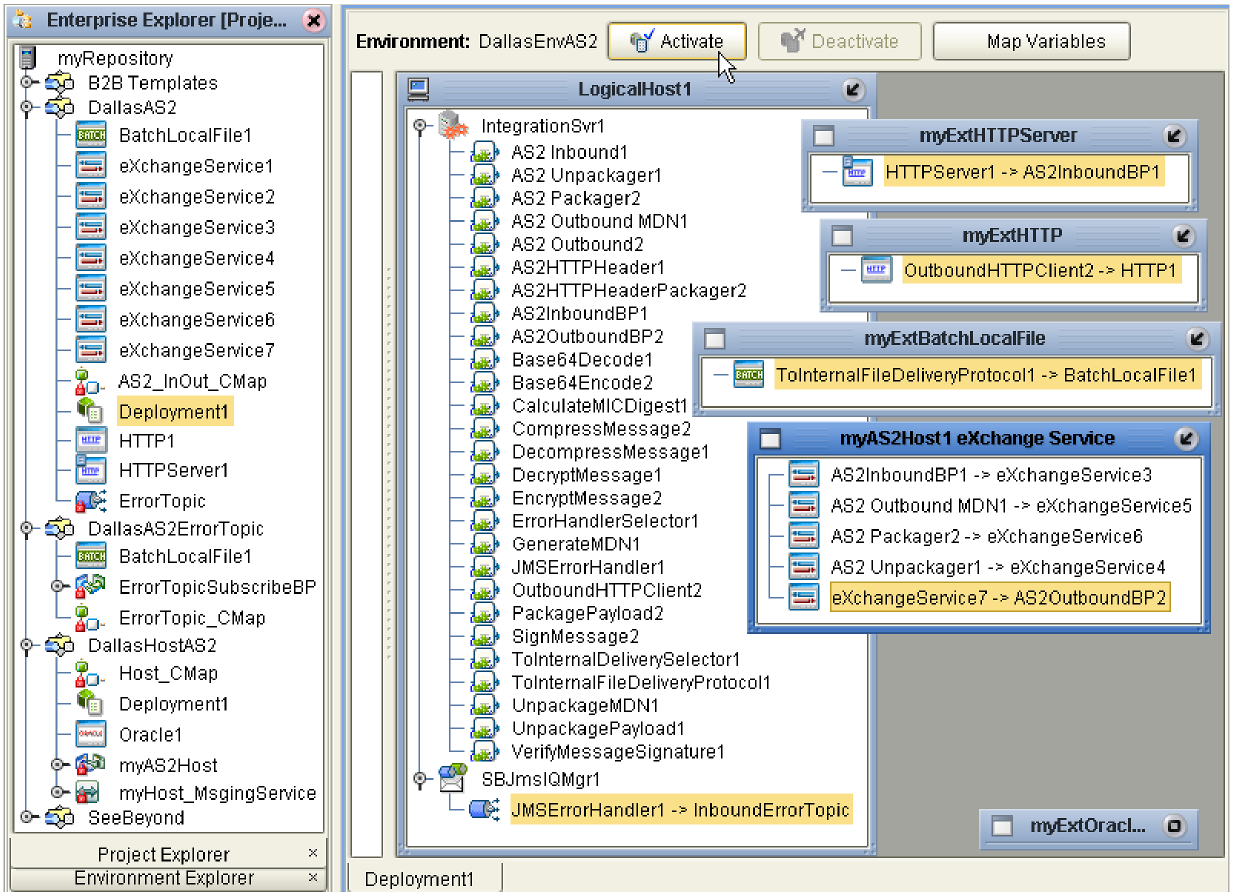
- DcpFinderCollab1 -> Oracle1
- PackMshSignalCollab1 -> Oracle1
- RetrieveAckCollab1 -> Oracle1
- TrackAckCollab1 -> Oracle1
- TrackerCollab1 -> Oracle1

BerlinHost1 eXchange Service

- ebXML Inbound1 -> eXchangeService1
- ebXML Outbound1 -> eXchangeService1

myExtFileIn

myExtFileOut



10 In response to the dialog box, click **No**; that is, do *not* apply to Logical Host(s).

To activate the ebXML error-handler project

Purpose: The purpose of the JmsErrorToFileReceiver project is to provide a durable subscriber for ErrorTopic in the ebXML_Protocol project, allowing Enterprise Manager to monitor messages in it. A simple B2B protocol, ErrorTopicSubscribeBP, receives its output and writes to C:\temp\eXchange\Sample\Data\Berlin\.

Before you begin: Your environment must be named Berlin_Env_ebXML (that is, it must correspond to the name of your host project), and it must contain an eXchange service; if necessary, see the **procedure on page 22**.

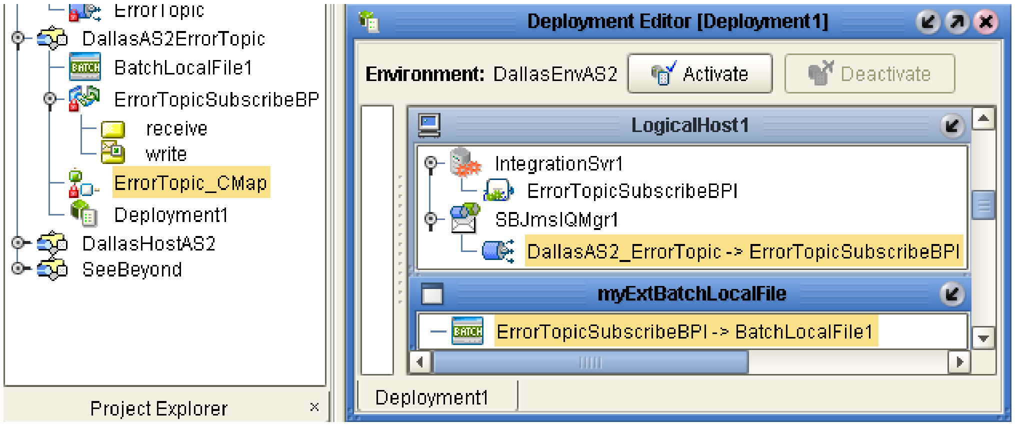
1 In the Project Explorer tree, right-click **JmsErrorToFileReceiver** and, on the popup context menu, point at New and click **Deployment Profile**

2 Keep the default name (Deployment1), point it at Berlin_Env_ebXML, and click OK.

3 Drag the BP service into LogicalHost1 and under IntegrationSvr1, and then drag the topic and under SbJmsIQMgr1.

4 Drag the BatchLocalFile eWay into myExtBatchLocalFile. See Figure 9.

Figure 9 Activating the ebXML Error-Handler Project



5 Click **Activate**. Save all, close canvases, and click **Refresh All from Repository**.

Result: All Berlin projects are activated, and will run when the logical host is started.

3.2.5. Setting Up the Atlanta (Company A) Projects

Purpose: Because this implementation demonstrates messages between two different B2B hosts, it requires a second complete setup. On the same machine or another, repeat all the preceding steps, starting from section 3.2.1, with the following differences:

- Wherever you see “Berlin”, substitute: “Atlanta”.
- In the Oracle properties, the users for Berlin and Atlanta must be different.
- Wherever a reference to company B appears, replace it with company A. For example, in section 3.2.3 **“Configuring the eXchange Service with Crypto Information” on page 23**, use **companya_pvtkey** instead of companyb_pvtkey.
- If the same physical machine will be running both logical hosts, ensure that the logical hosts use different base ports. See the **tip on page 20**.

3.3 Design Steps in ePM

For this sample implementation, design-time steps in eXchange Partner Manager (ePM) consist of the following:

- Importing the Berlin Trading Partner on page 28
- Configuring Trading Partner Parameters for the "Berlin" TP on page 29
- Activating the Trading Partner on page 31

3.3.1 Importing the Berlin Trading Partner

Before you begin: Your repository and your eXchange 5.0.4 Oracle database must be running and accessible. Enterprise Designer does ***not*** need to be running, and you do ***not*** need to have any logical hosts running.

To start eXchange Partner Manager (ePM)

1. Start a *new* browser session (that is, do ***not*** clone a new window of an existing session) pointing it at a repository URL, with **epm** appended. For example:

   - If your repository were running local on port 12000, the URL would be:

     `http://localhost:12000/epm`

   - For a repository running on machine herMachine on port 33000, it would be:

     `http://herMachine:33000/epm`

   - As usual, IP addresses are also permissible:

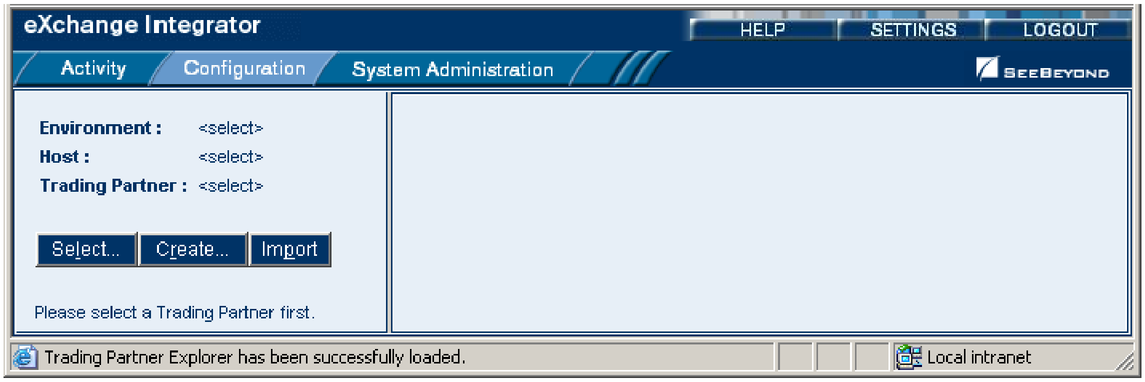
     `http://10.18.75.85:36271/epm`

   The string **epm** is case sensitive. In other words, ePM, Epm, and EPM are all errors.

2. When the sign-in screen appears, enter the Enterprise Manager username and password and click **Sign In**.

Result: The status bar (along the lower margin of the window) confirms that Trading Partner Explorer has loaded successfully, and the initial ePM screen appears, with no environment, host, or trading partner. See Figure 10.

Figure 10 Initial ePM Screen



Environment : <select>
Host : <select>
Trading Partner : <select>

Please select a Trading Partner first.

Trading Partner Explorer has been successfully loaded.

To import the “Atlanta” trading partner to the Berlin host

1. From the initial ePM screen, in the upper left side, click **Import**.

   A new window opens, prompting you to select a B2B host and trading partner.

2. Open the B2B Repository and **Berlin_Env_ebXML** and click **BerlinHost1**.
3. Enter **Atlanta**, browse to C:\temp\eXchange\Sample\TradingPartners and open **ebXML_BerlinHost_TP.xml**, and then click **Import**.

Result: In the explorer tree, under Berlin_Env_ebXML, new trading partner **Atlanta** appears.

To find the At trading partner

1. In the upper left side of the ePM screen, click **Select**.

   A new window opens, prompting you to select a B2B host and trading partner.

2. Open the B2B Repository and **Berlin_Env_ebXML** and click myAS2Host1.
3. Click **Search**, and then click OK.

Result: In the explorer tree, under Berlin_Env_ebXML, trading partner **Manhattan** reappears.

To import the “Berlin” trading partner to the Atlanta host

1. In the upper left side of the ePM screen, click **Import**.

   A new window opens, prompting you to select a B2B host and trading partner.

2. Open the B2B Repository and **Atlanta_Env-ebXML** and click **AtlantaHost1**.
3. Enter **Berlin**, browse to C:\temp\eXchange\Sample\TradingPartners and open **ebXML_AtlantaHost_TP.xml**, and then click **Import**.

Result: In the explorer tree, under Atlanta_Env_ebXML, new trading partner **Berlin** appears.

3.3.2 Configuring Trading Partner Parameters for the “Berlin” TP

When you imported the trading partner, parameter settings were valuated in part based on parameters stored in the export file, and in part based on the name of the trading partner. In this section, you will set or update the following:

- **Parameters for the Delivery Channel on page 29**
- **Parameters for the Internal Delivery Channel on page 30**

Parameters for the Delivery Channel

Purpose: To set the parameters governing Atlanta’s message exchange with “Berlin”. You are configuring a trading partner for the Atlanta environment, and so take the viewpoint of the Atlanta B2B host: “ToPartner” means “to Berlin”; “FromPartner” means “from Berlin”.

To configure the delivery channel parameters for trading partner “Berlin”

1 In the explorer (lower left) side of the ePM screen, click **Berlin**.

The canvas displays the trading partner’s general properties.

2 Click the **Components** tab.

The trading partner’s delivery channel parameters are displayed. See Table 5.

Table 5 Delivery Channel Parameters for Trading Partner “Berlin”

Binding Name	ebxml_delivery_channel1
ToPartner Transport Name	HTTP
FromPartner Transport Name	HTTP
Packager Name	

3 Click the binding name, **ebxml_delivery_channel1**.

The delivery channel’s general properties are displayed.

4 Click the **ToPartnerTransport** tab and edit the All Purpose End Point so that it contains the correct URL for Atlanta. For example:

```
http://localhost:19004/Deployment1_servlet/Inbound
```

If the logical host running the Atlanta B2B host is not named localhost, or if its integration server uses a port other than 19004 for its web server, or if the name of the deployment profile is not Deployment1, make the appropriate changes. When you are done, click **Save**.

5 Click the **FromPartnerTransport** tab and edit the All Purpose End Point so that it contains the correct URL for Berlin. For example:

```
http://localhost:18004/Deployment1_servlet/Inbound
```

If the logical host running the Berlin B2B host is not named localhost, or if its integration server uses a port other than 18004 for its web server, or if the name of the deployment profile is not Deployment1, make the appropriate changes. When you are done, click **Save**.

6 Click the **ToPartnerPackaging** tab, make changes as needed, and then click **Save**.

7 Click the **FromPartnerPackaging** tab, make changes as needed, and then click **Save**.

Result: For trading “Berlin”, the parameters for external delivery channel ebxml_delivery_channel1 are now set correctly.

Parameters for the Internal Delivery Channel

Purpose: To set the parameters governing Atlanta’s internal message processing when handling messages received from Berlin or preparing messages to be sent to Berlin. You are configuring a trading partner for the Atlanta environment, and so take the viewpoint of the Atlanta B2B host: “ToInternal” means “to the Atlanta internal (having been received from Berlin)”; “FromInternal” means “from the Atlanta internal (and destined for sending to Berlin)”.

Your configuration of the internal delivery channel depends on your setup.

Note: *Internal Delivery Channel bindings are always required for a project to run. Although it is possible to activate a trading partner whose messaging actions lack IDCs, the Channel Manager would have no instructions to read any input.*

3.3.3 Activating the Trading Partner

To activate the "Berlin" trading partner

Purpose: To save all the configuration information to the Oracle database to make it available at run time.

Before you begin: Your eXchange 5.0.4 Oracle database for the corresponding B2B host must be running.

1. In the explorer (lower left) side of the ePM screen, click **Berlin**.
2. In the bottom lower left of the the canvas, click the **Activate** button.
3. In response to the confirmation prompt, click **Activate**.

   The canvas displays a confirmation: **Trading Partner is successfully activated.**

Result: The **Berlin** trading partner is entirely complete and ready to be run.

Index