WebSphere® Adapters

Version 6.0.2

Adapter for Email User Guide

Note
Before using this information, be sure to read the general information in "Notices" on page 141.

22December2006

This edition applies to version 6, release 0, modification 2 of WebSphere Adapter for Email (product number 5724-N43) and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this document, email doc-comments@us.ibm.com. We look forward to hearing from you.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Chapter 1. About this information

This documentation is for integration developers who implement, configure, and deploy WebSphere® Adapter for Email. To use it, you should understand business integration concepts and possess certain technical skills.

Integration developers design, assemble, test, and deploy business integration solutions. This information is for those who are deploying WebSphere Adapter for Email in a solution that requires data exchange between enterprise information systems (EIS) and Java™ 2 Platform, Enterprise Edition (J2EE) applications. To use it, you should understand and have experience with the following concepts, standards, and tools:

- The business solution and environment.
- Databases, data access issues, transactional models, and connections across heterogeneous relational databases, queues, and Web services.
- Business integration mechanisms, including the Service Component Architecture (SCA) programming model and the Service Data Objects (SDO) data model.
- The J2EE standard and J2EE applications.
- The capabilities and requirements of WebSphere Process Server or WebSphere Enterprise Service Bus, depending on the host used in the environment. You should know how to configure and administer the host server and how to use the administrative console.
- The tools and capabilities provided by WebSphere Integration Developer. You should know how to use these tools to wire components and complete other integration tasks.

To complete the deployment, the should know how to perform the following tasks:

- Create required scripts, tools, and templates for both testing and deployment
- Resolve interdependencies between entities such as enterprise beans, workflows, and Web pages
- Write procedures to use database access logic efficiently
- Build data models for external data access tools
- Implement security measures

Chapter 2. Release notes

The release notes for WebSphere Adapter for Email, version 6.0.2 summarize new features and functions in this release and document any known workarounds.

Release notes for this adapter can be found at the following Web site:Release notes for WebSphere Adapter for Email

Chapter 3. Introduction to WebSphere Adapters

IBM® WebSphere Adapters make it possible for Java 2 Platform, Enterprise Edition (J2EE) components, such as new e-business applications, to communicate with resources on an enterprise information system (EIS). An EIS is the information infrastructure for an enterprise (for example, an enterprise resource planning [ERP] system).

A WebSphere adapter acts as an intermediary between the J2EE component and the EIS, so that the J2EE component does not need to understand the low-level API or data structures of the EIS.

WebSphere Adapters can be one of two types: application or technology.

- Application adapters connect to existing packaged applications (such as SAP Software, Siebel, PeopleSoft Enterprise, and JD Edwards EnterpriseOne) so that you can make use of data and services specific to the applications.
- Technology adapters provide connectivity to data through such technologies and protocols as relational databases, flat files, e-mail messages, and FTP.

As part of the WebSphere family of products, WebSphere Adapters work with WebSphere Integration Developer and either WebSphere Process Server or WebSphere Enterprise Service Bus.

- WebSphere Integration Developer is the tooling environment for the WebSphere adapters.

  You use WebSphere Integration Developer to assemble a module that is deployed on WebSphere Process Server or WebSphere Enterprise Service Bus. From within WebSphere Integration Developer, you import the adapter (which is packaged as a resource adapter [RAR] file) and connect to the EIS. The enterprise service discovery wizard of WebSphere Integration Developer looks for data and services on the EIS and creates the interface information needed to gain access to the data and services. Finally, WebSphere Integration Developer generates a module that includes the adapter and the interface information.
- WebSphere Process Server or WebSphere Enterprise Service Bus is the runtime environment for the WebSphere adapters.

  You deploy the module generated by WebSphere Integration Developer to one of the servers.

The generation and deployment of the module are illustrated in the following figure.

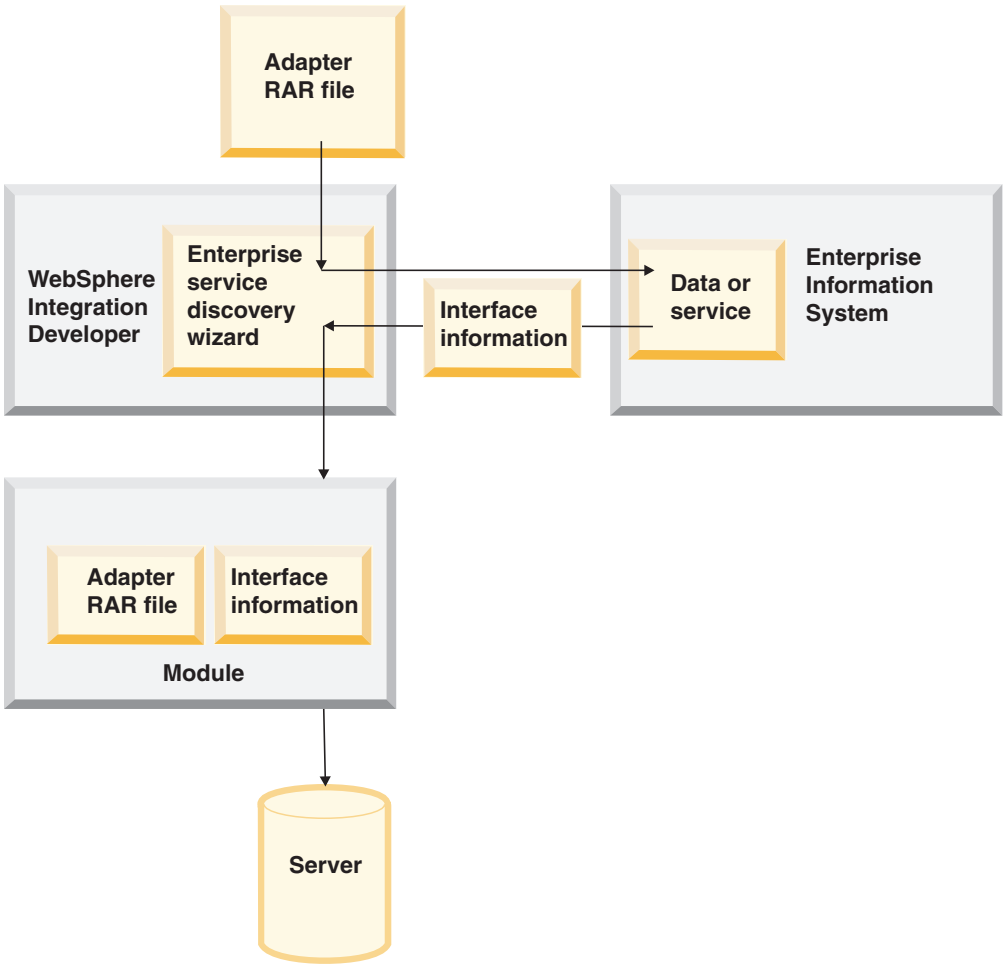


Figure 1. How a module is generated and deployed

Chapter 4. Introduction to the WebSphere Adapter for Email

The WebSphere Adapter for Email connects Java 2 Platform, Enterprise Edition (J2EE) components running on WebSphere Process Server or WebSphere Enterprise Service Bus with one or more mail servers. The adapter provides a means for the J2EE component and the mail server to interact.

Hardware and software requirements

Before installing, configuring, and using the adapter, you must understand the hardware and software environment that it requires. These requirements are available online.

View the hardware and software requirements for the WebSphere Adapter for Email on the IBM Web site at http://www.ibm.com/support/docview.wss?uid=swg27006249

Standards compliance

This product is compliant with several government and industry standards, including accessibility standards and Internet protocol standards.

Accessibility

IBM strives to provide products with usable access for everyone, regardless of age or ability. The WebSphere Adapters software is fully accessible and section 508-compliant. Accessibility features enable users with physical disabilities, such as restricted mobility or limited vision, to operate software products successfully. These features are built into the installation and administration features of WebSphere Adapters.

Installation

You can install WebSphere Adapters either through a graphical user interface or silently through a script. The silent installation method is recommended for users with accessibility needs.

Administration

The administrative console of either WebSphere Process Server or WebSphere Enterprise Service Bus is the primary interface for deployment and administration of the enterprise applications. These consoles are displayed within a standard Web browser. By using an accessible Web browser, such as Microsoft® Internet Explorer or Netscape Browser, you are able to:

- Use screen-reader software and a digital speech synthesizer to hear what is displayed on the screen
- Use voice recognition software, such as IBM ViaVoice®, to enter data and to navigate the user interface
- Operate features by using the keyboard instead of the mouse

You can configure and use product features by using standard text editors and scripted or command line interfaces instead of the graphical interfaces that are provided.

When appropriate, the documentation for specific product features contains additional information about the accessibility of the features.

Enterprise service discovery wizard

The enterprise service discovery wizard is the primary component used to create enterprise applications with the adapters. This wizard is implemented as an Eclipse plug-in that is available through WebSphere Integration Developer and is fully accessible.

Keyboard navigation

This product uses standard Microsoft Windows® navigation keys.

IBM and accessibility

See the *IBM Accessibility Center* for more information about the commitment that IBM has to accessibility.

Internet Protocol Version 6.0

IBM WebSphere Process Server relies on WebSphere Application Server for Internet Protocol Version 6.0 compatibility.

IBM WebSphere Application Server Version 6.0 and its JavaMail component support dual stack Internet Protocol Version 6.0 (IPv6).

For more information about this compatibility in WebSphere Application Server, see IPv6 support in the WebSphere Application Server information center.

For more information about IPv6, see www.ipv6.org.

Technical overview of the Adapter for Email

WebSphere Adapter for Email enables enterprise information system (EIS) connectivity across e-mailing systems. Because the adapter is equipped to handle both inbound and outbound communication with an application server, it can retrieve an e-mail from a mail server based on selective search criteria and then use the information in e-mails to trigger an automated flow of operation. Similarly, it can send a response to the e-mail address or addresses that initiates the operation.

The adapter integrates EISs by sending and receiving e-mails to or from different mail servers using SMTP, IMAP, and POP3 e-mail protocols. The interaction between the mail server and the adapter is handled through the JavaMail API provided by the IBM Java development kit.

In inbound operational mode, the adapter:

1. Polls the mail server at a user-configured interval
2. Reads the e-mail retrieved from the inbox
3. Converts the e-mail into a business object
4. Sends the business object to the application server

In outbound operational mode, the adapter:

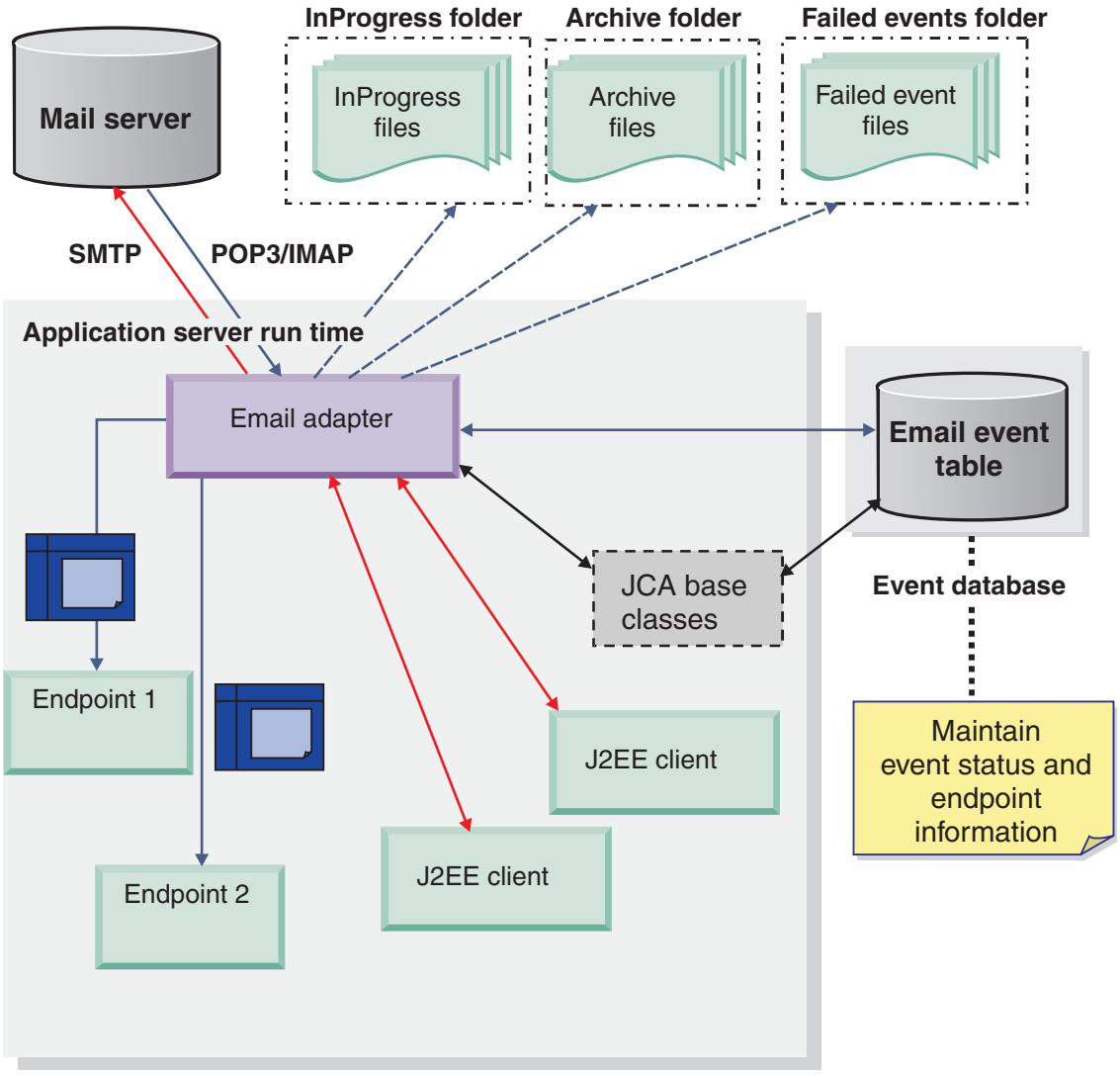
1. Receives a business object from the application server

2. Converts the business object to an e-mail
3. Sends the e-mail to the mail server.

Adapter architecture

As noted in the figure below, the Email adapter is housed within the application server run time. The adapter facilitates communication between a mail server and various intelligent programs identified in the figure as end points using information stored in business objects. To keep track of endpoints and status, the adapter maintains an event table.

Email adapter architecture



Outbound processing

When the application server sends a business object to the adapter, the adapter processes the business object and generates an e-mail.

For the adapter to process outbound events successfully, its business objects must contain to address and from address information. They may also contain MIME type information, business objects, content, and subject information. To include a file as an e-mail attachment, you can specify the path in the business object to

where the adapter can find the file. A single business object can result in an e-mail message containing multiple business object strings and multiple file attachments.

Note: The adapter supports synchronous outbound messages.

To process a business object request, the adapter takes the following actions:

1. Extracts metadata information (to and from addresses, encoding and mime-type information) from the top level business object and composes an e-mail message.
2. Processes contained business objects as follows:
   - Converts each business object into a business object string and includes it as an attachment to the e-mail. The MIME type of the attachment is defined in the top level business object. There is only one business object per attachment.
   - If there are file paths specified in the top level business object, the adapter includes the referenced files as attachments to the e-mail. Each file must be located at its specified location.
3. Delivers the mail using the SMTP protocol to the configured mail server.

If the adapter cannot create the e-mail, it logs a failure exception.

Inbound processing

The adapter polls the mail server at a specified interval, retrieves new messages from specified mail server folders, converts each message to a business object, and then sends the business objects to the application server.

Required folders for inbound communication

The following folders need to be present for each poll location. If you set up the adapter to poll in multiple locations, you must have these folders in each location.

Required folders:

- In progress folder - Located on your local drive, it holds e-mails that are marked in progress in the event table.
- Poll folder - One or more folders on the mail server that the adapter polls for new e-mails.

Supported inbound protocols

Depending on your mail server, you might be able to pick which inbound e-mail protocol to use. Differences between the protocols and a description of how the adapter processes each protocol are noted in the following table and sections that follow.

Table 1. Differences between the IMAP and POP3 protocols

IMAP	POP3
Supports the existence of multiple mail folders on a mailbox.	Supports only one mailbox (named ″Inbox″) per user.
Allows a copy of the e-mail to remain on the mail server after the client receives the e-mail.	Supports a ″View-once-only″ feature on the server. The mail is deleted from the mail server once the client receives a copy of the mail.

Inbound with IMAP

If you use the IMAP protocol for your inbound communication, the adapter performs the following steps:

1. The adapter polls the mail folders at regular intervals and logs any unread e-mails as events in the event table. You can use the activation specification PollFolders property to customize the list of folders that are searched.
2. If you specify multiple mail server folders for the mail server account in the PollFolders property, the adapter polls all mail server folders sequentially.
3. Search criteria determines which mail is picked up from the mail server. The adapter picks up all e-mail that match the criteria
4. The adapter writes all polled e-mails to the InProgress folder in RFC822 format using the Message-ID of the polled e-mail as the file name. Once an e-mail is written into the InProgress folder, it is deleted from the mail server.
5. The adapter transforms each e-mail message into a business object instance and delivers it to the enterprise application.
6. The adapter deletes processed e-mails from the InProgress folder and archives them if archiving is selected.

Inbound with POP3

If you use the POP3 protocol for your inbound communication, the adapter performs the following steps:

1. The adapter polls the Inbox mail folder at regular intervals and logs any unread e-mails as events in the event table.
2. If you specify search criteria, all unread e-mails that fit the search criteria are picked up for polling.

   Note: When no search criteria is specified, all unread mail is picked up by default.
3. The adapter writes all polled e-mails to the InProgress folder in RFC822 format and uses the Message-ID of the polled e-mail as the file name.
4. Once an e-mail is read by the adapter it is deleted.
5. The adapter transforms each e-mail message into a business object instance and delivers it to the enterprise application.
6. The adapter deletes all processed e-mail from the InProgress folder and archives them if archiving is selected.

Archiving

You can configure the adapter to archive both successful and failed e-mails. To do this, the following folders must be present:

- Archive folder – A file system folder where the adapter archives successful events.
- Failed event folder – A file system folder where the adapter archives failed events.

When the ArchiveFolder property is specified, all successfully processed mail will be moved into the archive folder from the in progress folder. If you leave this property blank all successfully processed mail is deleted from the InProgress folder.

If the FailedEventsFolder is specified, all unsuccessfully processed mail will be moved into the failed events folder from the in progress folder. If you leave this property blank, all failed mail is deleted from the in progress folder.

Event store

The adapter uses event tables to keep track of e-mails as they pass through the application server. Each inbound instance of the adapter must have it's own event table. Multiple instances of the adapter cannot point to the same event table.

When the adapter polls the mail server, it creates an entry in an event table for each e-mail that matches the search criteria. The adapter records the status of each new entry as NEW. As the adapter copies the e-mail from the mail server to a local folder, it marks the entry as IN PROGRESS. When the adapter sends the event out to the function selector for data transformation, it deletes the entry from the event table.

Business objects

Business objects carry the functional properties, data transformation information, and mail content that the adapter needs to process requests and generate e-mails. Depending on your business need, you can use the default business objects created by the enterprise service discovery wizard along with imported custom business object definitions (xsd files) that you have stored on your local system.

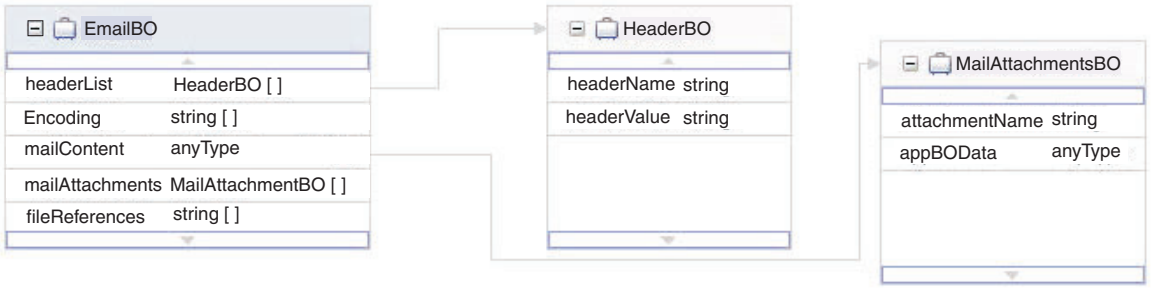
Business object structure

The adapter defines and generates three default business objects during enterprise service discovery– EmailBO, HeaderBO, and MailAttachmentBO. The EmailBO business object is a wrapper business object that contains the HeaderBO and MailAttachmentBO business objects as children. These default objects are the same for both inbound and outbound communication.

EmailBO

The EmailBO business object contains all of the details required by the adapter to process inbound and outbound events. As shown in the following figure, it is a parent object that contains the business object structures for the mail headers, mail content, and mail attachments as children.

Note: To use custom headers, you must note them in the `headerList` property of the EmailBO object.



EmailBO business object structure

HeaderBO

Each HeaderBO business object consists of a header name and a header value. The header business object is constructed dynamically at runtime, after the adapter is deployed.

The HeaderBO business object stores all standard (RFC822) e-mail headers and any custom user headers defined in the EmailBO business object.

Note: The Bcc and Resent-bcc header values cannot be retrieved by the adapter from an e-mail message.

MailAttachmentBO

Each MailAttachmentBO business object consists of an attachment name and one e-mail attachment's worth of data. Each e-mail attachment is stored in its own MailAttachmentBO object.

During inbound processing, the attachments are parsed and the contents sent out as business objects. One attachment gets parsed into one MailAttachmentBO business object.

During outbound processing, the application server sets the data within the MailAttachmentBO business object during the request. The data in this business object then becomes an e-mail attachment when the e-mail message is created by the adapter.

Note: MailAttachmentBO objects can store e-mail attachments of any user-defined type (such as Customer or PurchaseOrder).

Custom business objects

If you choose to use custom business objects, you must create pre-defined business objects using WebSphere Integration Developer business object wizard before running enterprise service discovery. The business object definitions created by the wizard are stored as xsd files on your local system. When enterprise service discovery wizards creates business objects, the wizard will look for the pre-defined business objects created in the business object wizard and populate them with adapter project-specific data.

For more information on how to create pre-defined business objects, see the WebSphere Integration Developer documentation.

Business object naming convention

Business object names must reflect the structure they represent, such as Customer or Address. Names will most likely be derived during the metadata import process of enterprise metadata discovery, based on the name given by the mail server.

Business object names must be converted to camel case, in which separators such as spaces and underscores are removed, and the first letter of each word is capitalized; for example, ORDER_LINE_ITEM would be converted to OrderLineItem.

The parent business object graph must be named for the contained business object, followed by BG; for example, CustomerBG for a Customer business object.

Business object names have no semantic value to the adapter or the database.

Business object verb support

For outbound communication, the adapter supports the verb Create. When the create operation is invoked, the adapter creates an e-mail and posts it to the mail server. There are no supported verbs for inbound communication.

Business object application-specific information

The adapter stores data transformation information at two levels of business object level application-specific information. At the WrapperBO business object level, the adapter stores the information necessary to match content types to their appropriate data binding implementation. At the ApplicationBO business object level, the adapter stores the WrapperBO business object name, as well as content type association and encoding information.

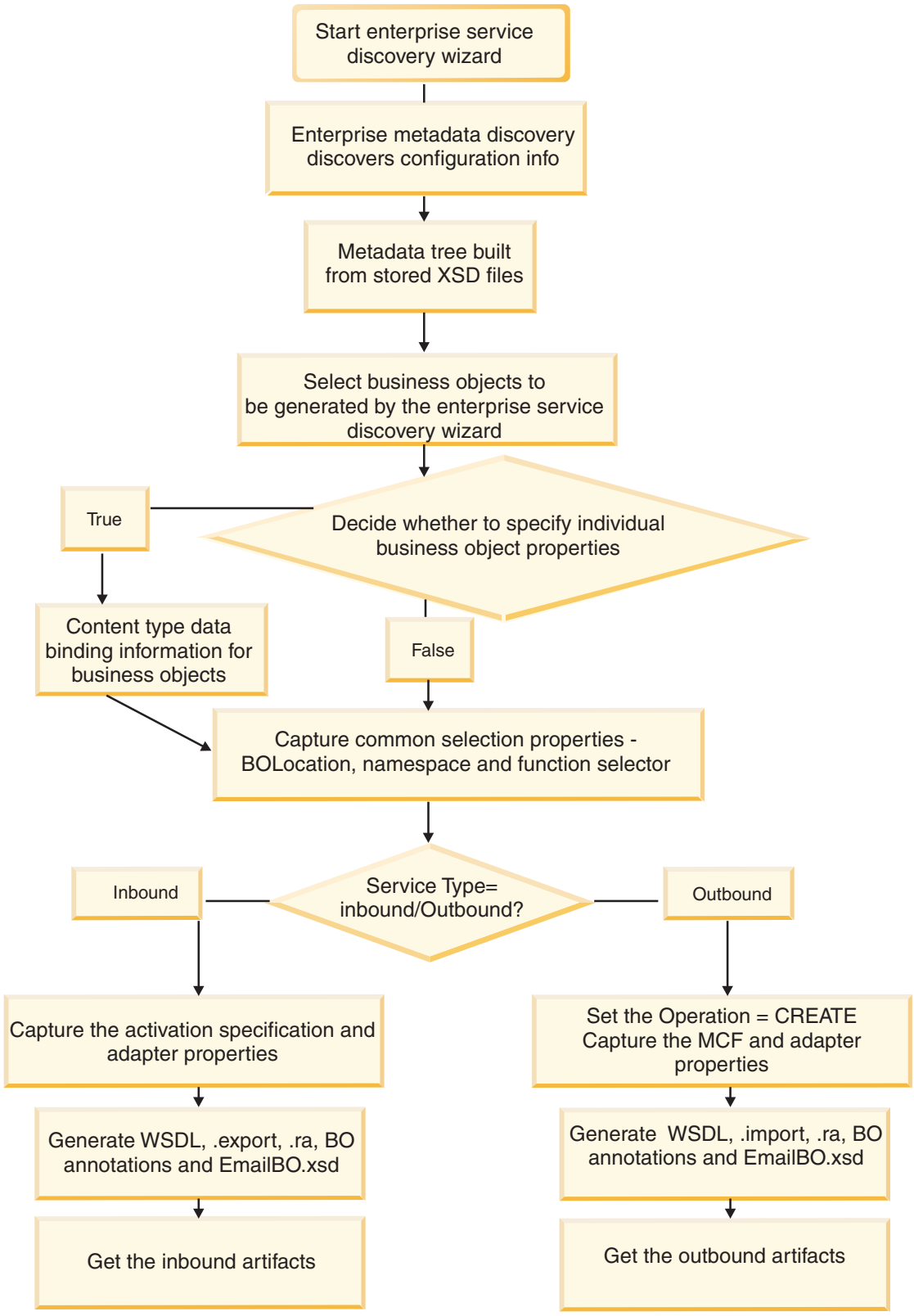
Enterprise service discovery

The enterprise service discovery wizard provides a blueprint for creating business objects. By selecting meta-object nodes from the metadata tree structure in the wizard, you can generate business objects for your mail server, build service descriptions, create business objects, and apply application-specific information and properties to the business objects.

With the enterprise service discovery wizard, you can perform the following actions:

- Generate business objects
- Set application-specific information on business objects
- Set application-specific information on properties
- Provide service descriptions for inbound and outbound events
- Provide connection descriptions for inbound and outbound events

The following figure illustrates the enterprise service discovery wizard flow. When finished, an EAR file containing all of the information for your adapter project is created. This EAR file can then be deployed to the application server.



Basic enterprise service discovery wizard flow

Globalization and bidirectional transformation

The adapter is globalized to support single- and multi-byte character sets and deliver message text in the specified language. The adapter also performs bidirectional transformation, which refers to the task of processing data that contains both left-to-right (Hebrew or Arabic, for example) and right-to-left (a URL or file path, for example) semantic content within the same file.

Globalization

The Java runtime environment within the Java virtual machine (JVM) represents data in the Unicode character code set. Unicode contains encodings for characters in most known character code sets (both single- and multi-byte). Components in the WebSphere Business Integration system are written in Java. Therefore, when data is transferred between WebSphere Business Integration system components, there is no need for character conversion.

To log error and informational messages in the appropriate language and for the appropriate country or region, the adapter uses the locale of the system on which it is running.

Bidirectional transformation

Languages such as Arabic and Hebrew are written from right to left, yet they contain embedded segments of text that are written left to right, resulting in bidirectional script. When software applications handle bidirectional script, standards are used to display and process it. WebSphere Process Server and WebSphere Enterprise Service Bus use the Windows standard format, but an enterprise information system exchanging data with WebSphere Process Server or WebSphere Enterprise Service Bus can use a different format. WebSphere Adapters transform bidirectional script data passed between the two systems so that it is accurately processed and displayed on both sides of a transaction.

Bidirectional format

WebSphere Process Server and WebSphere Enterprise Service Bus use the bidirectional format of ILYNN (implicit, left-to-right, on, off, nominal). This is the format used by Windows. If an enterprise information system uses a different format, the adapter converts the format prior to introducing the data to WebSphere Process Server or WebSphere Enterprise Service Bus.

Five attributes comprise bidirectional format. When you set bidirectional properties, you assign values for each of these attributes. The attributes and settings are listed in the following table.

Table 2. Bidirectional format attributes

Letter position	Purpose	Values	Description	Default setting
1	Order schema	I or V	Implicit (Logical) or Visual	I
2	Direction	L R C D	Left-to-Right, Right-to-Left Contextual Left-to-Right Contextual Right-to-Left	L
3	Symmetric Swapping	Y or N	Symmetric Swapping is on or off	Y

Table 2. Bidirectional format attributes (continued)

Letter position	Purpose	Values	Description	Default setting
4	Shaping	S N I M F B	Text is shaped Text is not shaped Initial shaping Middle shaping Final shaping Isolated shaping	N
5	Numeric Shaping	H C N	Hindi Contextual Nominal	N

The adapter transforms data into a logical, left-to-right format before sending the data to WebSphere Process Server or WebSphere Enterprise Service Bus.

Using bidirectional properties

You can use multiple bidirectional properties to control the transformation of both content data and metadata. You can set special bidirectional properties to exclude either content data or metadata from bidirectional transformation, or to identify data that requires special treatment during a transformation.

The following table describes four types of bidirectional properties.

Table 3. Bidirectional property types

Property type	Data transformations
EIS	Controls the format for content data, or data that is sent by the enterprise information system.
Metadata	Controls the format for metadata, or data that provides information about the content data.
Skip	Identifies content or metadata to exclude from transformation.
Special Format	Identifies certain text, such as file paths or URLs, that require different treatment during the transformation process. Can be set for either content data or metadata.

You can set properties that control bidirectional transformation in three areas.

- **Resource adapter properties:** These properties store default configuration settings, including the TurnBiDiOff property, which controls whether the adapter instance performs bidirectional transformation or not. Use the administrative console of the server to configure these properties.
- **Managed (J2C) connection factory properties:** These properties are used at run time to create an outbound connection instance with an enterprise information system. Once the managed connection factory properties are created, they are stored in the deployment descriptor.
- **Activation specification properties:** These properties hold the inbound event processing configuration information for a message endpoint. Set them as you perform enterprise service discovery, or use the administrative console of the server.

Business object annotations

Some adapters allow you to annotate bidirectional properties within a business object. Do this to add information that specifically controls the transformation of a business object or part of a business object. Use business object editor, a tool within WebSphere Integration Developer, to add annotations at these levels:

- Business object
- Business object application-specific attribute
- Business object attribute
- Business object attribute application-specific attribute

Property scope and lookup mechanism

After you set values for bidirectional properties for an adapter and annotate business objects where appropriate, the adapter performs bidirectional transformations. It does so by using logic that relies on a hierarchical inheritance of property settings and a lookup mechanism.

Properties defined within the resource adapter are at the top of the hierarchy, while those defined within other areas or annotated within a business object are at lower levels of the hierarchy. So for example, if you only set values for EIS-type bidirectional properties for the resource adapter, those values are inherited and used by transformations that require a defined EIS-type bidirectional property whether they arise from an inbound (activation specification) transaction or an outbound (managed connection factory) transaction.

However, if you set values for EIS-type bidirectional properties for both the resource adapter and the activation specification, a transformation arising from an inbound transaction uses the values set for the activation specification.

The processing logic uses a lookup mechanism to search for bidirectional property values to use during a transformation. The lookup mechanism begins its search at the level where the transformation arises and searches upward through the hierarchy for defined values of the appropriate property type. It uses the first valid value it finds. It searches the hierarchy from child to parent only; siblings are not considered in the search.

Chapter 5. Planning for adapter implementation

Before you begin your installation, you must consider several factors, such as your adapter environment, security and performance needs, and whether you need locale and globalization support.

Security

The adapter is Java 2 security enabled and features username and password authentication. In addition, you can configure additional security permissions by altering the application server's WAS.policy file and storing it in the meta-inf folder. For more details on configuring security details, see the security documentation for WebSphere Process Server.

WebSphere Adapters in clustered environments

You can improve adapter performance and availability by deploying the WebSphere adapter enterprise archive (EAR) module to a clustered server environment. The adapter instance within the EAR module is replicated across federated servers.

WebSphere Process Server and WebSphere Application Server Network Deployment support clustered environments. Clusters are groups of servers that are managed together to balance workloads and to provide high availability and scalability. When you set up a server cluster, you create a Deployment Manager profile. The HAManager, a subcomponent of the Deployment Manager, notifies the JCA container to activate the adapter instance. The JCA container provides a runtime environment for adapter instances. For more information about clustered environments, see http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm_cluster_v61.html.

In clustered environments, adapter instances can handle both inbound and outbound operations.

High availability for inbound operations

Inbound operations are based on events triggered as a result of updates to data in the enterprise information system (EIS) application. The adapter is configured to detect updates through event listeners or by polling an event table. The adapter then publishes the event to its endpoint.

In a clustered environment, two or more adapter instances might detect the same event. This scenario raises the possibility of duplicate event processing or data infidelity. For example, if two adapter instances are simultaneously polling the same event table with the same event type filter, one may alter data that the other adapter instance depends on, or it might fail. There is a parallel risk for event-listening adapter architectures in a clustered environment.

To avoid this condition, the HAManager for the inbound adapter instances enforces a singleton behavior. Even though all the adapter instances are started, only one of the instances detects and publishes an event to the endpoint for each type of EIS application.

When you deploy an adapter module to a cluster, the JCA container checks the enableHASupport property of the ResourceAdapter bean. If the value for the enableHASupport property is true, the JCA container registers all of the adapter instances with HAManager with a policy 1 of N. This policy means that only one of the clustered servers starts event polling (or listening) for this adapter instance. Although other adapter instances in the cluster are started, they remain dormant with respect to the active event until the active adapter instance finishes processing the event. If the server on which the polling thread was started shuts down for some reason, an adapter instance that is running on one of the backup servers is activated.

High availability for outbound operations

In clustered environments, multiple adapter instances are available to perform outbound requests. Accordingly, if your environment has multiple applications that interact with the same WebSphere adapter for outbound requests, then you might improve performance by deploying the adapter module to a clustered environment.

WebSphere Application Server Network Deployment has a workload management capability that distributes the outbound processing among the adapter instances. As a result, outbound operations in a clustered environment are similar to those in a single server environment: one adapter instance processes only one outbound request at a time. For more information on workload management, see http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm.html.

Note: Adapter instances are replicated in a clustered server environment. When the enableHASupport property is set to true, which is the default setting, only one of the replicated adapter instances actively polls for events while other instances are in standby mode. If the enableHASupport property is set to false, all of the adapter instances replicated on cluster members actively poll for events. This may result in event duplication. Do not change the value of enableHASupport to false for single server environments. For information on changing the value of this property, see the Resource adapter properties section in this documentation. To determine whether adapter replication is supported in a clustered environment, see the software and hardware requirements section of this documentation.

Roadmap for installing, configuring, and deploying the adapter

Before you can use the adapter in a runtime environment, you must install, configure, and deploy it. Understanding these tasks at a high level helps you perform the steps that are needed to accomplish each task.

After successfully installing the WebSphere Adapter, you configure it using WebSphere Integration Developer. You then deploy it as an enterprise archive (EAR) file to WebSphere Process Server or WebSphere Enterprise Service Bus. The following figure illustrates this flow of tasks, and the steps that follow the figure describe each task at a high-level. For detailed instructions on installing, see *Installing IBM WebSphere Adapters*. For information about configuring and deploying the adapter, see the adapter documentation.

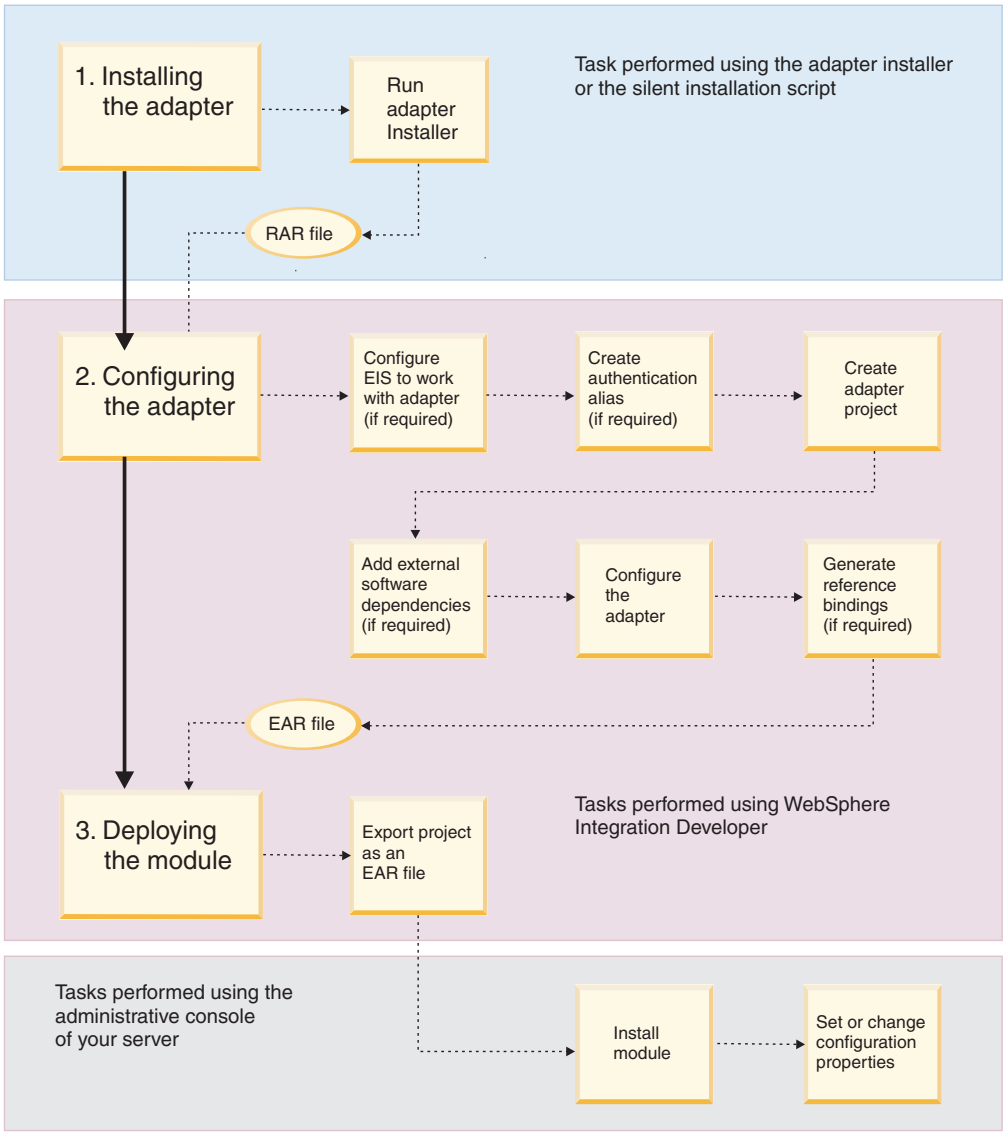


Figure 2. Roadmap for installing, configuring, and deploying the adapter

1. **Installing the adapter**
   a. Use the installer (a graphical user interface) or a script that runs a silent installation. Either method installs a resource adapter archive (RAR) file on your workstation. You use this RAR file to configure the adapter.
2. **Configuring the adapter**
   a. (If required) Configure the enterprise information system (EIS) to work with your adapter. You perform this step from within the EIS application.
   b. (If required) Create an authentication alias to access the application.
   c. Create an adapter project in WebSphere Integration Developer (J2EE Perspective) by importing the adapter RAR file.
   d. (If required) Using WebSphere Integration Developer, add any external dependencies required by your adapter to the adapter project. These dependencies are also required as part of the bundled EAR file, which is exported when you deploy the adapter.

e. To configure the adapter, run the enterprise service discovery wizard from the Business Integration Perspective of WebSphere Integration Developer. The enterprise service discovery wizard generates business integration components and allows you to enter all the information necessary to configure the adapter for the first time. The output from the enterprise service discovery tool is saved to a business integration module project, which contains the business object, or objects, and the import or export file.

f. (If required) Use WebSphere Integration Developer to generate reference bindings for the component created by the enterprise service discovery wizard.

3. **Deploying the module**

   a. From the J2EE perspective in WebSphere Integration Developer, export a business integration module project as an EAR file.

   b. Install the module on WebSphere Process Server or WebSphere Enterprise Service Bus.

   c. (If required) In the server administrative console, set (or change) the following properties:

      - Resource adapter properties
      - Managed (J2C) connection factory properties
      - Activation specification properties for the EIS

Chapter 6. Installing WebSphere Adapter for Email, version 6.0.2

To install the adapter, you must check system prerequisites, then perform the installation steps common to all adapters.

Installation prerequisites

Before installing Adapter for Email, you must verify that your environment meets all of the necessary hardware and software requirements. These requirements fall into two categories: supported platforms for running the adapter installer, and hardware and software requirements for configuring, deploying, and running the adapter.

Supported platforms for running the adapter installer

The supported platforms for running the adapter installer are located in the ″Installing″ section of Installing IBM WebSphere Adapters.

Hardware and software requirements for configuring, deploying, and running the adapter

The hardware and software requirements for configuring, deploying, and running the adapter are located at the following Web site: IBM WebSphere Adapters and IBM WebSphere Business Integration Adapters: software requirements. From the IBM WebSphere Adapters list, select the link for the Adapter for Email, Version 6.0.2.

Additional jar files

If you are using WebSphere Integration Developer version 6.0.1.1 or earlier, you must manually add three additional jar files to the classpath of the connector project. For more information on how to do this, see ″Adding jar files to WebSphere Integration Developer versions 6011 and earlier″ in the reference section.

Performing the installation

The steps for installing the adapter are the same for all WebSphere Adapters. You can install the adapter either by using a graphical user interface or by performing a silent installation.

Before you begin

Review the installation prerequisites.

How to perform this task

1. Install the adapter using the basic installation instructions, which are common to all adapters. These steps are located in the ″Installing″ section of Installing IBM WebSphere Adapters.

Note: Some WebSphere Adapters require you to perform additional steps specific to your adapter to complete the installation. The WebSphere Adapter for Email does not have this requirement.

2. After performing the basic installation steps, you can configure the adapter.

Result

The resource adapter archive (RAR) file is copied to the workstation where the adapter is installed. If you accepted the default installation location, the RAR file is placed in the following directory: C:\Program Files\IBM\ResourceAdapters\Email\adapter\Email\deploy\CWYEM_Email.rar.

What to do next

Configure the adapter.

Uninstalling the adapter

The steps for uninstalling the adapter are the same for all WebSphere Adapters. You can uninstall the adapter either by using a graphical user interface or by performing a silent uninstallation.

About this task

Uninstalling the adapter may be a required task for troubleshooting an installation problem. The steps for uninstalling the adapter are located in the ″Uninstalling″ section of Installing WebSphere Adapters.

Note: If you need to uninstall an adapter that is already deployed, refer to the ″Additional adapter-related information you might need″ section of “Related product information” on page 137.

Chapter 7. Configuring the adapter for deployment

Once you have installed the adapter, use WebSphere Integration Developer enterprise service discovery wizard to configure the adapter.

Creating the authentication alias

Create the authentication alias on the server by using the WebSphere Process Server administrative console. From the administrative console, configure the global security and set the password for the authentication alias, which is used to process outbound requests.

Before you begin

You must create an authentication alias on the server if you have not done so already.

About this task

To create an authentication alias, use the following procedure.

How to perform this task

1. On the WebSphere administrative console "Welcome page," click **Security** → **Global security**.
2. Under the Authentication heading, click **JAAS Configuration** → **J2C Authentication data**.
3. Click **New**.
4. Type the required information in the **Alias**, **User ID**, **Password**, and **Description** fields.

   Note: The user ID and password that you type will be used to establish a connection to the enterprise information system for outbound processing.
5. Click **OK**, click **Save**, and then click **Save** again.

Configuring the EIS to work with the adapter

In an Email adapter implementation, the EIS is your mail server. Before you create your adapter project, you must install your mail server and configure authentication details. You must also create a folder or folders on the mail server for the Email adapter to poll for inbound events and add the Java mail and Activation jar files to the adapter runtime classpath.

Creating the adapter project

To begin the process of creating a module to communicate with an Email service, you create an adapter project. The adapter project contains the adapter itself plus other related artifacts. You create the project by importing the RAR file, which was copied to your local file system during installation, into WebSphere Integration Developer.

About this task

To create an adapter project, use the following procedure.

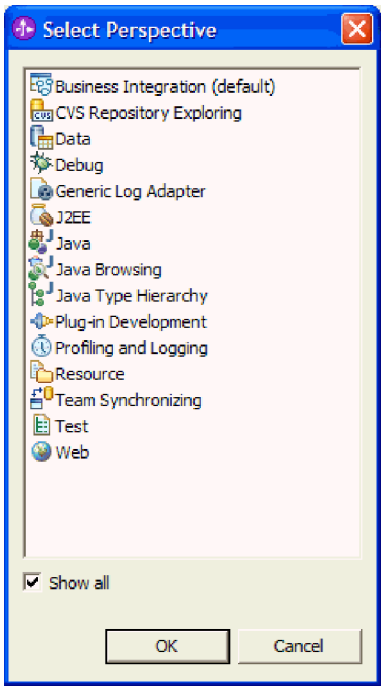
How to perform this task

1. Launch WebSphere Integration Developer by clicking **Start** → **Programs** → **IBM WebSphere** → **Integration Developer 6.0** → **WebSphere Integration Developer 6.0**.
2. If you are prompted to specify a workspace, accept the default value.

   The workspace is a directory where WebSphere Integration Developer stores your project.
3. When the WebSphere Integration Developer window is displayed, close the Welcome page.
4. Switch to the J2EE perspective:
   a. Click **Window** → **Open Perspective** → **Other**.
   b. Click **J2EE**.

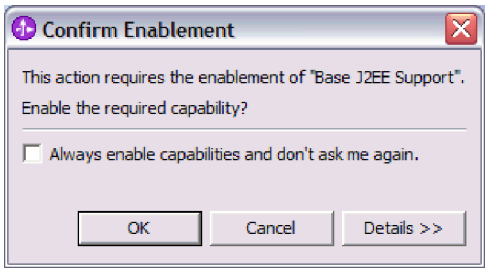
      If **J2EE** is not displayed, select the **Show all** check box, click **J2EE**, and click **OK**.

Figure 3. Selecting J2EE from the Select Perspective list



   c. If you see the Confirm Enablement window, select **Always enable capabilities and don't ask me again**.

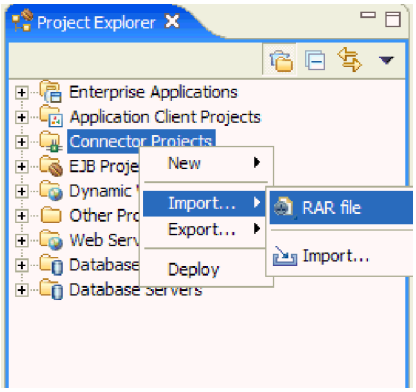
Figure 4. The Confirm Enablement window



d. Click **OK**.

5. Import the RAR file by right-clicking **Connector Projects** and clicking **Import → RAR file**.

Figure 5. Importing the RAR file

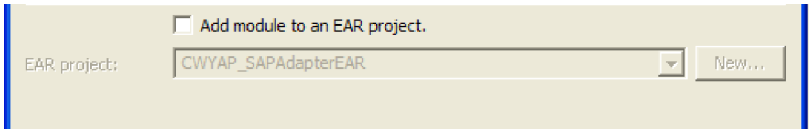


6. Find the RAR file on your local file system by clicking **Browse** and navigating to the directory in which Adapter for Email was installed.
7. Accept the default setting (**CWYEM_Email.rar**) for **Connector project**.

   The connector project has the same name as the RAR file.
8. Accept the default value in the **Target server** field.

   The default value is the test environment for WebSphere Process Server, which is installed as part of WebSphere Integration Developer.
9. Clear the **Add module to an EAR project** check box.

Figure 6. Clearing the Add module to an EAR project check box



   Notice that the EAR project field becomes unavailable after you remove the check mark.
10. Click **Finish**.

Result

The RAR file is imported and a connector project is created. The connector project is visible in the Project Explorer window under Connector Projects.

What to do next

Configure the adapter for outbound processing.

Creating custom data bindings

The adapter provides an option for creating custom data bindings to meet your environment needs. A custom data binding is a Java class that you write to convert a stream of data to a business object during inbound processing and a business object to a stream of data during outbound processing. The enterprise information system (EIS) and JMS exports and imports can be configured to invoke custom data bindings.

About this task

The following steps guide you through the process of creating custom data bindings.

How to perform this task

1. Implement the custom data binding using the commonj.connector.runtime.RecordHolderDataBinding interface. This interface is located in the commonj.connector.jar file, which is contained in the WebSphere Process Server runtime libraries.
2. Create the custom data bindings using either the inbound or outbound information shown below.
   - To convert a stream of data to a business object for inbound processing, the adapter calls the public void setRecord(Record arg0) and public DataObject getDataObject() methods of the custom data binding. It first calls setRecord and then calls getDataObject.
     a. The setRecord method takes an com.ibm.j2ca.base.UnstructuredRecord as a parameter. This class is part of the Adapter Foundation class jar file. The business object record, as text or byte[], is sent to the custom data binding in this method. Other required information is also sent.
     b. The getDataObject() gets the business object record, set as mentioned above, does the necessary data transformation and returns the required business object.
     c. The getRecordName() present in the unstructured record gets the fully qualified namespace with which we can instantiate the business object and extract (as mentioned in step 3) any required ASI present in the annotation. This ASI is used by the data binding. For example:

        ```
        recordName = http://www.ibm.com/xmlns/prod/websphere/j2ca/ftp/
        customerwrapperbg/CustomerWrapperBG

        namespace = http://www.ibm.com/xmlns/prod/websphere/j2ca/ftp/
        customerwrapperbg

        object name = CustomerWrapperBG
        ```

   - To convert a business object to a stream of data for outbound processing, the adapter calls the public void setDataObject(DataObject dataObject) and public Record getRecord() methods of the custom data binding. It first calls setDataObject and then calls getRecord.

a. The setDataObject method takes the business object as a parameter. The output of getRecord must be of type com.ibm.j2ca.base.UnstructuredRecord.
b. Application-specific information is obtained from the input business object as mentioned in step 3.
c. The getRecord method does the necessary transformation and returns the native data in the form of an unstructured record.

An example of EIS native data with delimited text is:
Employee~Create~sarath~IBM~Bangalore~560071

Business object schema:

```
complexType name="Employee"
   sequence maxOccurs="1" minOccurs="1"
    element maxOccurs="1" minOccurs="0" name="name" type="string"/
    element maxOccurs="1" minOccurs="0" name="company" type="string"/
    element maxOccurs="1" minOccurs="0" name="city" type="string"/
    element maxOccurs="1" minOccurs="0" name="zip" type="string"/
   /sequence
  /complexType
```

3. To access application-specific information in the annotation of the business object, use the CWYBS_AdapterFoundation.jar file, which contains a class named com.ibm.j2ca.base.AdapterBOUtil. This class contains the method to extract the application-specific information from the business object.
4. Use the business object application programming interface (API) methods to extract each value contained in the annotation. The output is a business object representing the application-specific information.
   - For inbound, instantiate the business object from the `recordName` value contained in the unstructured record.
   - For outbound, call the `getMetadataForObject()` by passing the business object as a parameter.

What to do next

Configure the custom data bindings using the detailed instructions located in *WPS_install_directory*\samples\doc\CustomEISDataBinding\index.html.

Configuring the adapter for outbound processing

To configure WebSphere Adapter for Email for outbound processing, use the enterprise service discovery wizard in WebSphere Integration Developer to set the connection properties for enterprise service discovery, select business objects or services that are in the enterprise information system, and generate business object definitions and related artifacts for outbound processing.

Generating business objects using enterprise service discovery

Use the enterprise service discovery wizard to generate your outbound business objects. In the wizard, set the connection properties so the wizard knows where to look for services and business objects, select business objects, and then generate your business object definitions and artifacts.

Setting connection properties for enterprise service discovery

Use enterprise service discovery to determine which services are available to the adapter and to configure the adapter's mail server connection settings.

Before you begin

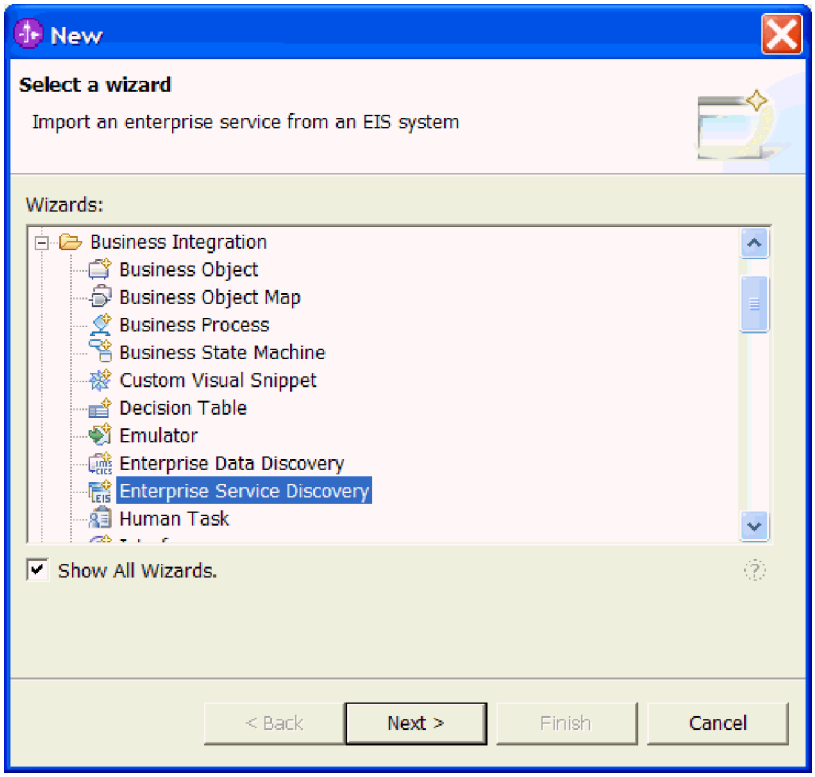
Before you can set the connection properties in this section, you must have created your adapter project.

About this task

To set enterprise service discovery connection properties, follow this procedure. See "Enterprise service discovery connection properties" on page 117 for more information about these properties.

How to perform this task

1. Change to the Business Integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration**.
2. Select **File** → **New** → **Enterprise Service Discovery**.
3. **Optional:** If Enterprise Service Discovery is not an option, complete the following steps.
   a. Select **File** → **New** → **Other**.
   b. Select the **Show all wizards check box** and then expand **Business Integration**.
   c. Select **Enterprise Service Discovery** and then click **Next**.

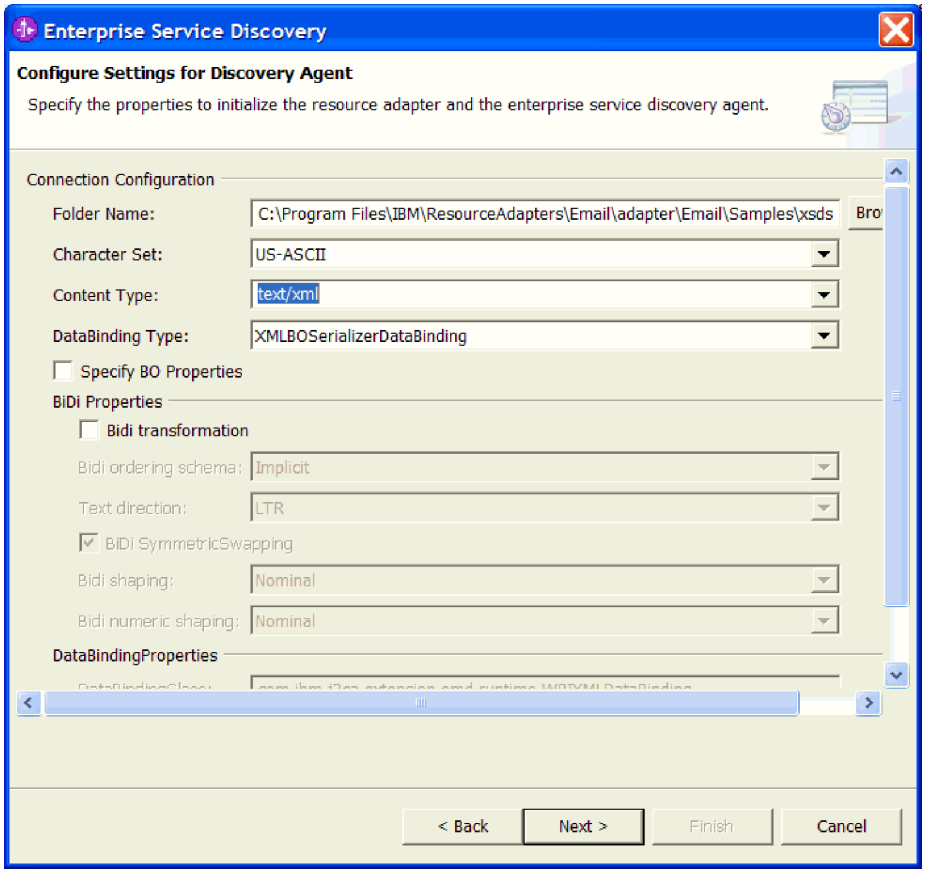


Select a wizard window

4. Select your connector project and click **Next**.
5. On the Configure settings for Discovery Agent window make the following selections:
   a. Browse for any pre-defined xsd files.

      Note: For a pass-through scenario, leave this field blank.

b. Define a **Character Set** for the adapter to use.
c. Select a **Content Type** format. When the content type is selected, the corresponding data binding types fields are populated.
d. **Optional:** To use **BiDi transformation** select the check box. When selected, more BiDi properties can be defined.
e. **Optional:** To change the log and trace file directory, click the **Show Advanced** button and browse for a new directory.



Configure settings for Discovery Agent window

f. Select **Next**.

Result

Using these properties, the enterprise service discovery wizard will initialize the adapter and the enterprise service discovery agent.

What to do next

Select the business objects and services to associate with the adapter module.

Selecting business objects and services to be used with the adapter

Use the enterprise service discovery tool to select business objects and services for use with your adapter.

Before you begin

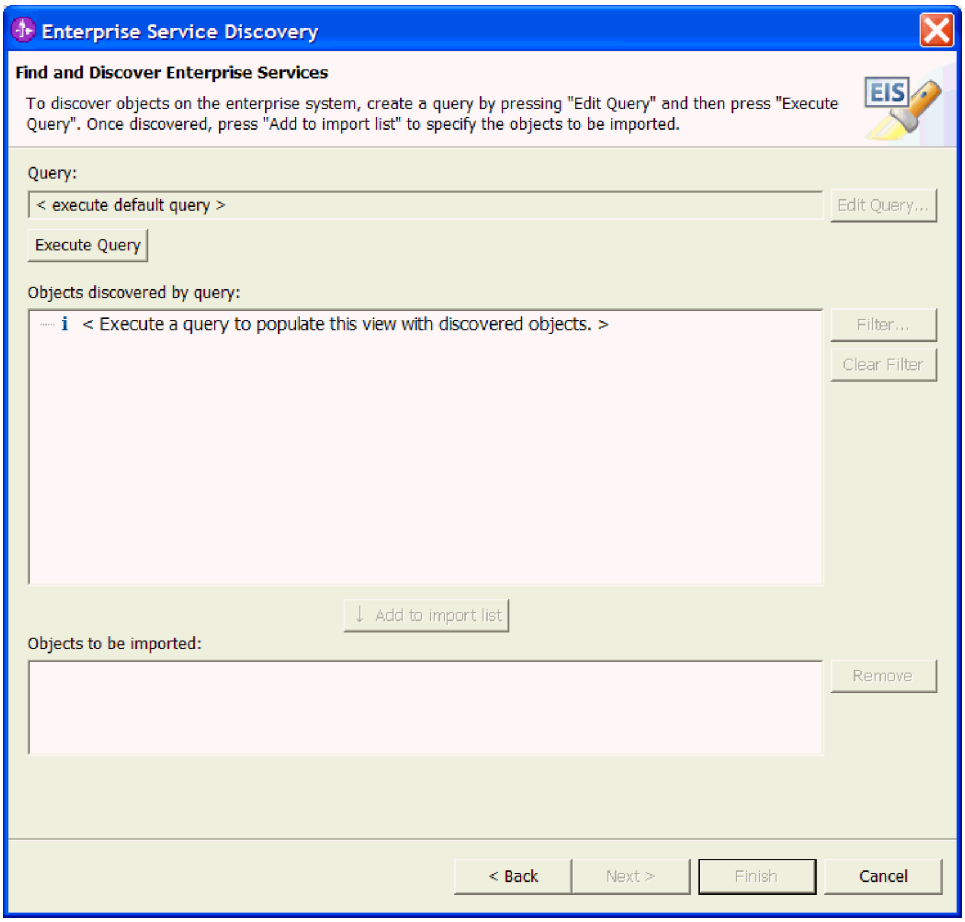
You must have specified the adapter configuration properties for the discovery service before you can complete the steps below.

About this task

To select business objects and services to be used with the adapter, follow this procedure.

How to perform this task

1. Click **Execute Query**. Any business objects found in the folder defined on the last screen are displayed under Objects discovered by query.
2. Select objects to add to the module and click **Add to import list**.



Find and Discover Enterprise Services window

3. When all of the desired business objects are listed under Objects to be imported, click **Next**.

Result

Using these properties, the discovery service prepares a metadata tree that will be displayed for object selection and navigation in the following steps.

What to do next

Configure the selected business objects.

Configuring the selected objects

Once you have added business objects to the module, configure them for outbound operations.

Before you begin

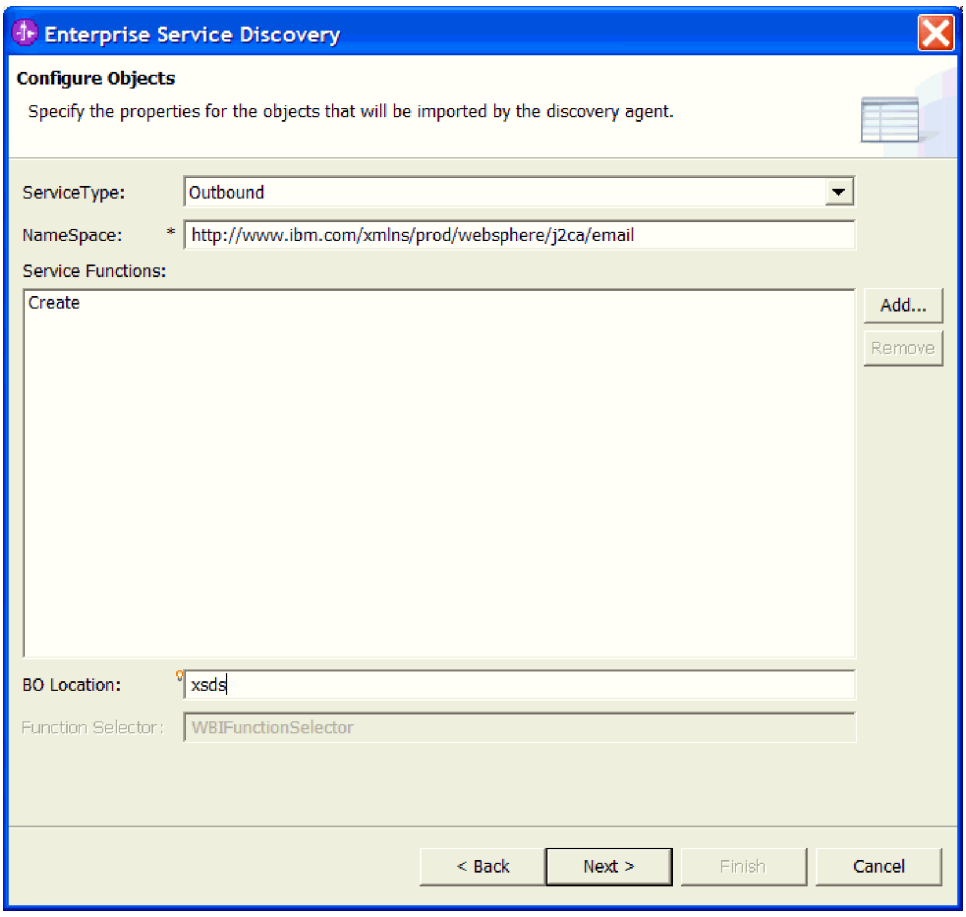
You must have added business objects to the adapter project.

About this task

To configure the selected business objects, follow this procedure. See "Enterprise service discovery selection properties" on page 118 for more information about these properties.

How to perform this task

1. Select **Outbound** from the **Service Type** drop down menu.
2. Leave the **NameSpace** value unchanged.
3. Specify the **BO Location** for your predefined xsd files and click **Next**.



Configure Objects window

Result

The objects have been configured for outbound communication.

What to do next

Generate outbound artifacts for the adapter project.

Generating artifacts

Define your connector project's business object definitions and their related artifacts.

Before you begin

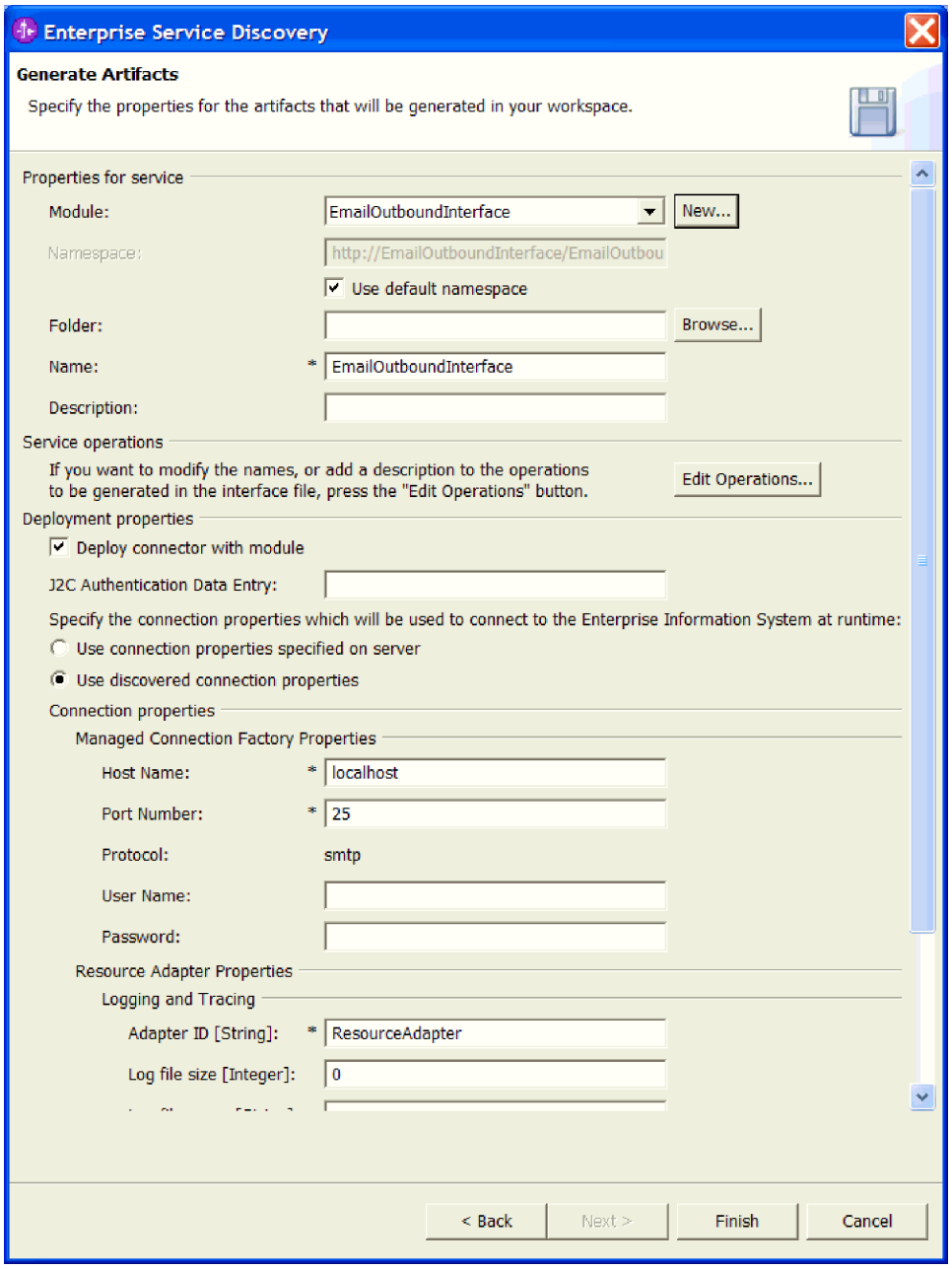
To generate business objects and artifacts for your connector project, you must have already selected business objects to add to the adapter project.

About this task

To generate artifacts for the adapter project, follow this procedure. See "Managed (J2C) connection factory properties" on page 121 for more information about these properties.

How to perform this task

1. Next to the **Module** field, click **New**. The New Integration Project window opens.
2. Select **Create a module project** and click **Next**.
3. Name the adapter module and click **Finish**. For outbound communications, consider naming the module EmailOutbound. The module is created and a new screen opens to define the managed connection factory properties.
4. **Optional:** To modify the Service operations names or description, click **Edit Operations**.
5. Leave the **Deploy connector with module** check box selected and click **Use discovered connection properties**.
6. **Optional:** Change the default **Host** name and **Port** values to reflect your mail server setup. For more information about event persistence properties, see "Managed (J2C) connection factory properties" on page 121
7. **Optional:** Change the **Username** and **Password** values to reflect your mail server setup. The UserName and Password entered should be valid for the e-mail server running on the specified host. For more information about event persistence properties, see "Managed (J2C) connection factory properties" on page 121
8. **Optional:** Change the **Log file name** and **Trace file name** paths to store the log and trace files in a different location.



Generate Artifacts window

9. Click **Finish** when all of the values have been entered.

Result

The enterprise service discovery wizard will generate the artifacts. The outbound artifacts that are created are visible in the WebSphere Integration Developer Project Explorer under your module.

What to do next

Generate reference bindings for the adapter module.

Generating reference bindings

Create a reference in the assembly editor from the adapter project to a stand-alone reference. The stand-alone reference represents an a generic J2EE component, such as the application server. By wiring the adapter project to the stand-alone reference, you link the adapter to other server processes.

Before you begin

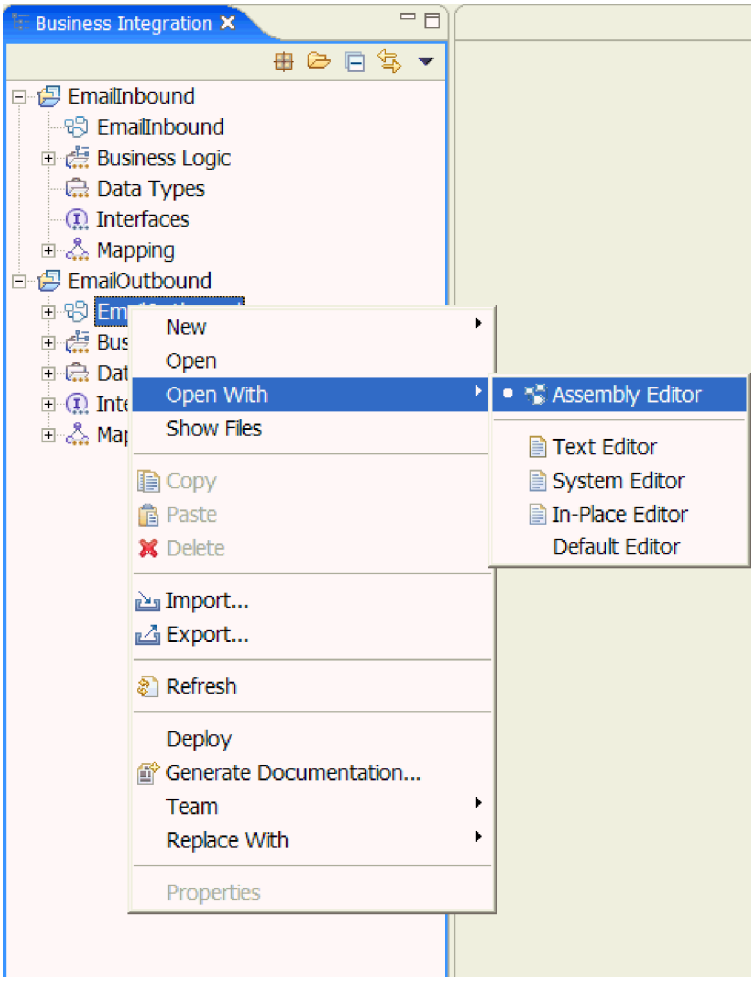
You must have generated artifacts for your adapter project before creating reference bindings.

About this task

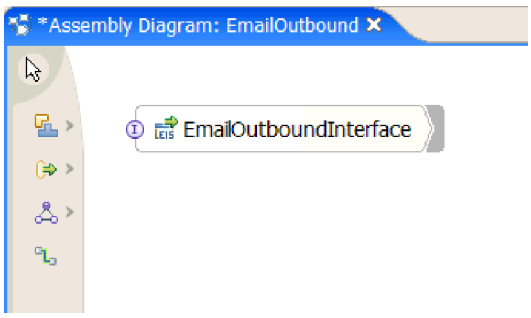
To generate reference bindings, follow this procedure. For more information on wiring reference bindings, see the WebSphere Integration Developer documentation.

How to perform this task

1. **Optional:** Switch to the business integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration Perspective** .
2. Right-click the outbound project and select **Open With** → **Assembly Editor** .

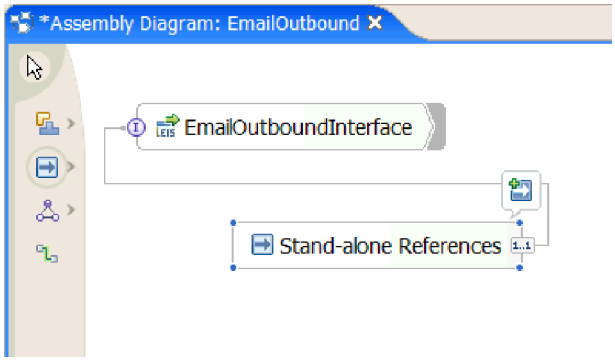


Opening the Assembly Editor



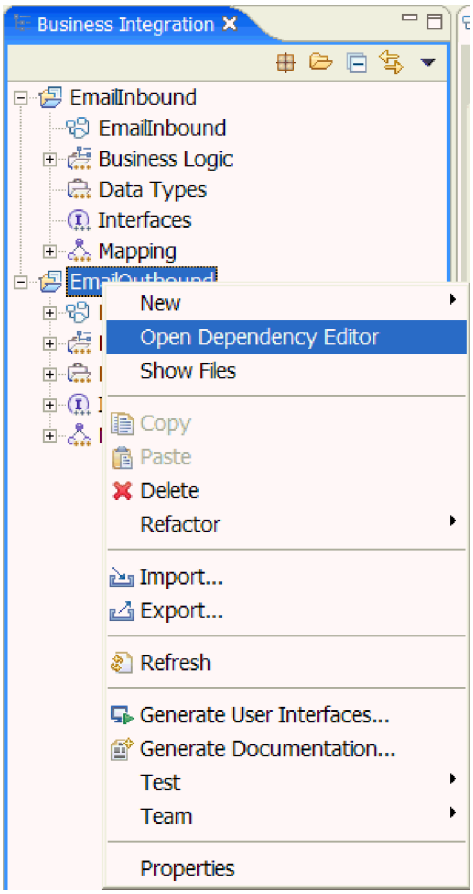
Assembly Diagram window with EmailOutboundInterface representation

3. In the assembly editor, locate and click the **Import** icon (a green arrow).
4. Locate and click the **Stand-alone References** icon (a blue arrow) once. Then, click once in the assembly editor to place the stand-alone reference.
5. Create a wire between the stand-alone reference to the adapter project by clicking the side of the stand-alone reference representation and dragging the wire to the adapter project representation.



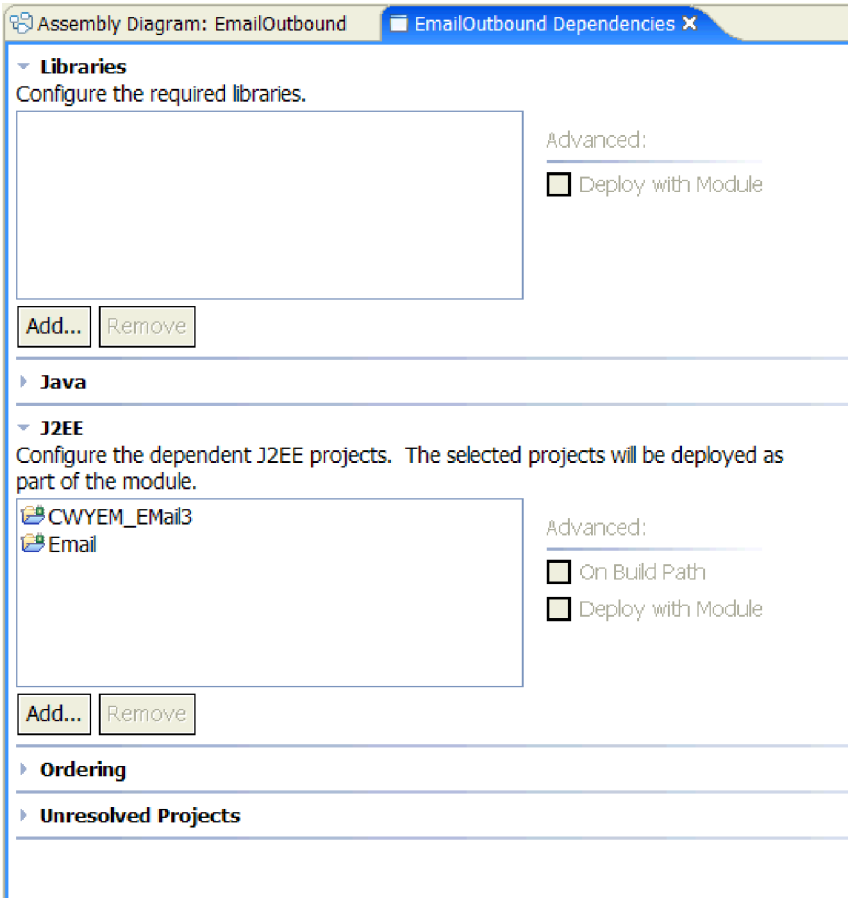
Assembly Diagram EmailOutboundInterface wired to Stand-alone References

6. **Optional:** If a message appears telling you that a reference will be added to the source node and asking you to continue, click **OK**.
7. Save the changes.
8. Right-click the adapter project and select **Open Dependency Editor**.



Opening Dependency Editor

9. Click **Add** to add your J2EE project.



Selecting a J2EE project

10. Select your project from the J2EE list and save.

Result

A reference binding is created linking the adapter to the J2EE client.

What to do next

Export the adapter project to an EAR file. The EAR file can be deployed to the application server.

Configuring the adapter for inbound processing

To configure WebSphere Adapter for Email for inbound processing, use the enterprise service discovery wizard in WebSphere Integration Developer to set the connection properties for the adapter, select business objects or services that are in the enterprise information system, and generate business object definitions and related artifacts for inbound processing.

Generating business objects using enterprise service discovery

Use the enterprise service discovery wizard to generate your inbound business objects. In the wizard, set the connection properties so the wizard knows where to look for services and business objects, select business objects, and then generate your business object definitions and artifacts.

Setting connection properties for enterprise service discovery

Use enterprise service discovery to determine which services are available to the adapter and to configure the adapter's mail server connection settings.

Before you begin

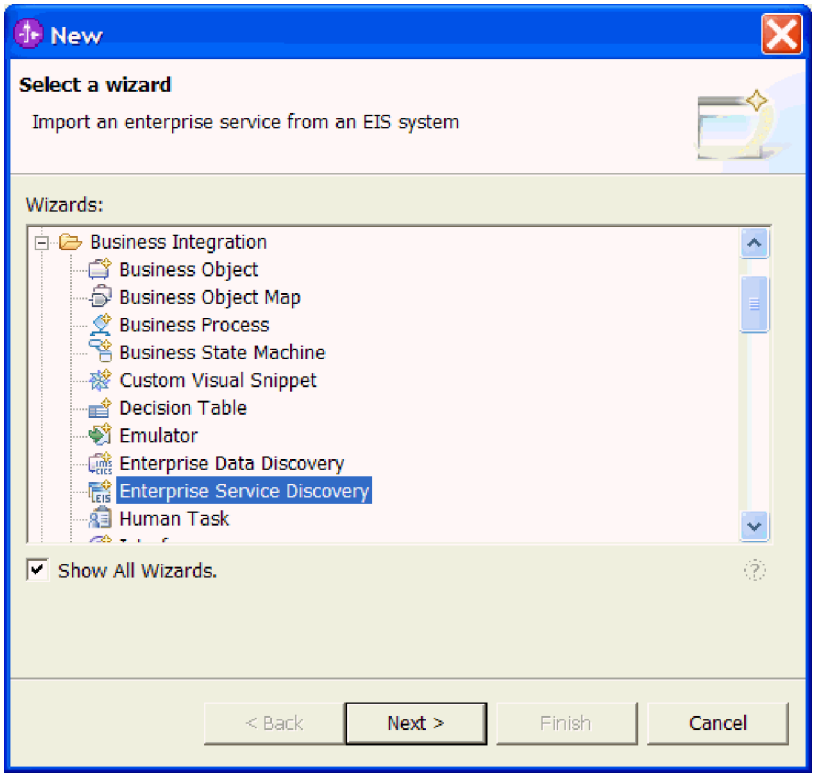
Before you can set the connection properties in this section, you must have created your adapter project.

About this task

To set enterprise service discovery connection properties, follow this procedure. See "Enterprise service discovery connection properties" on page 117 for more information about these properties.

How to perform this task

1. Change to the Business Integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration**.
2. Select **File** → **New** → **Enterprise Service Discovery**.
3. **Optional:** If Enterprise Service Discovery is not an option, complete the following steps.
   a. Select **File** → **New** → **Other**.
   b. Select the **Show all wizards check box** and then expand **Business Integration**.
   c. Select **Enterprise Service Discovery** and then click **Next**.



Select a wizard window

4. Select your connector project and click **Next**.
5. On the Configure settings for Discovery Agent window make the following selections:

a. Browse for any pre-defined xsd files.

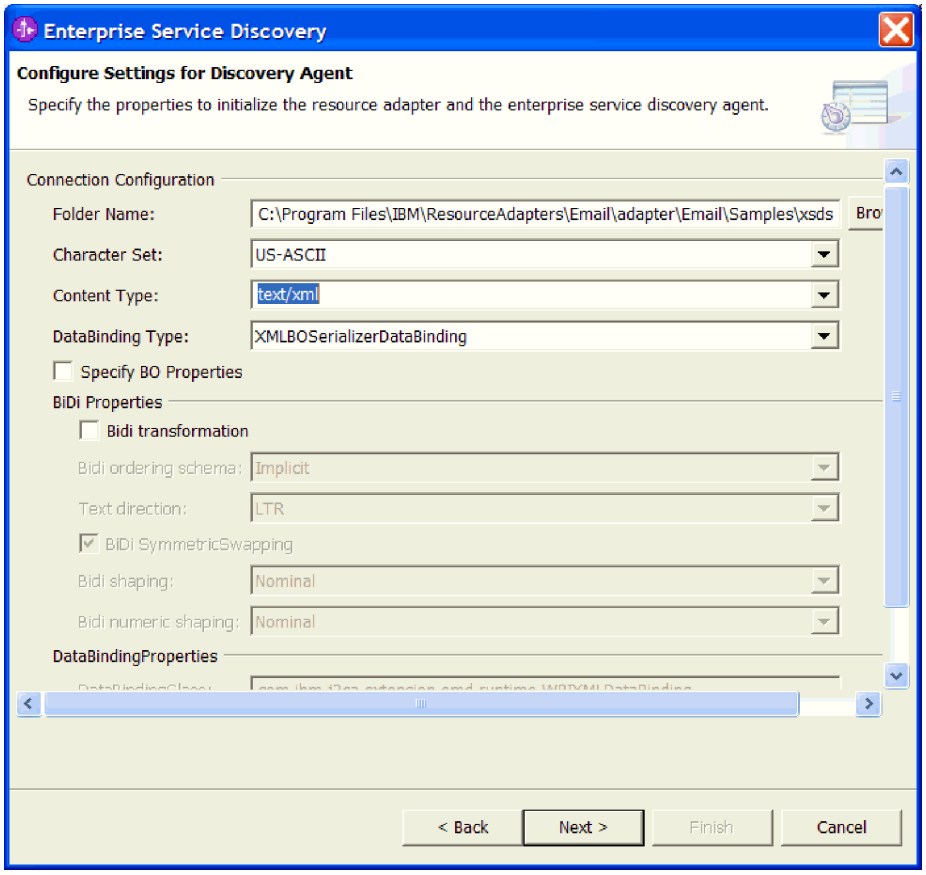
   Note: For a pass-through scenario, leave this field blank.

b. Define a **Character Set** for the adapter to use.

c. Select a **Content Type** format. When the content type is selected, the corresponding data binding types fields are populated.

d. **Optional:** To use **BiDi transformation** select the check box. When selected, more BiDi properties can be defined.

e. **Optional:** To change the log and trace file directory, click the **Show Advanced** button and browse for a new directory.



Configure settings for Discovery Agent window

f. Select **Next**.

Result

Using these properties, the enterprise service discovery wizard will initialize the adapter and the enterprise service discovery agent.

What to do next

Select the business objects and services to associate with the adapter module.

Selecting business objects and services to be used with the adapter

Use the enterprise service discovery tool to select business objects and services for use with your adapter.

Before you begin

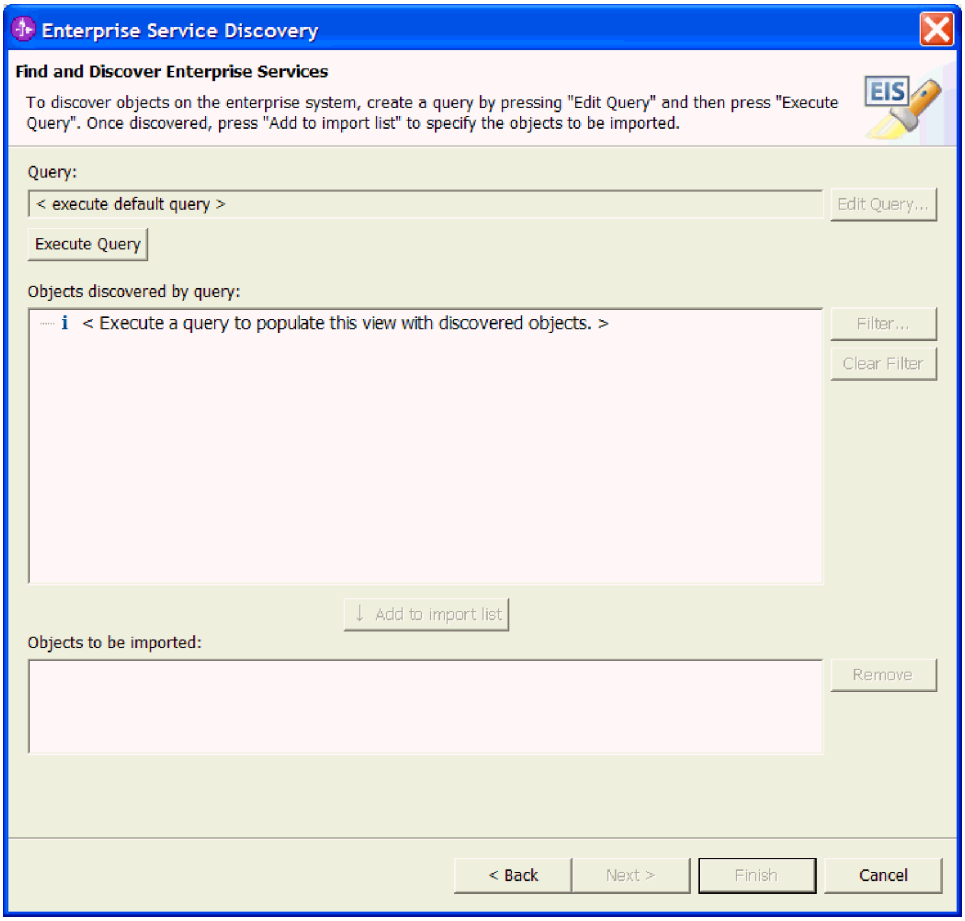
You must have specified the adapter configuration properties for the discovery service before you can complete the steps below.

About this task

To select business objects and services to be used with the adapter, follow this procedure.

How to perform this task

1. Click **Execute Query**. Any business objects found in the folder defined on the last screen are displayed under Objects discovered by query.
2. Select objects to add to the module and press **Add**.



Find and Discover Enterprise Services window

3. When all of the desired business objects are listed under Objects to be imported, click **Next**.

Result

Using these properties, the discovery service prepares a metadata tree that will be displayed for object selection and navigation in the following steps.

What to do next

Configure the selected business objects.

Configuring the selected objects

Once you have added business objects to the module, configure them for inbound operations.

Before you begin

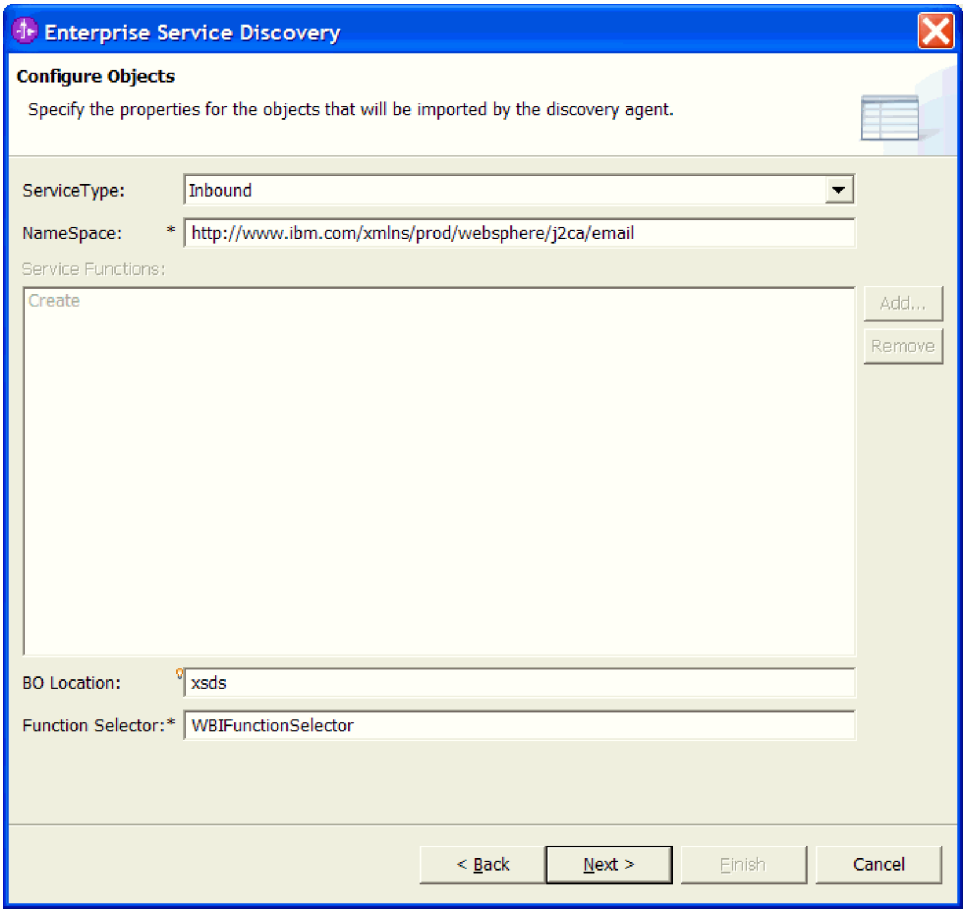
You must have added business objects to the adapter project.

About this task

To configure the selected business objects, follow this procedure. See "Enterprise service discovery selection properties" on page 118 for more information about these properties.

How to perform this task

1. Select **Inbound** from the **Service Type** drop down menu.
2. Leave the **NameSpace** value unchanged.
3. Specify the **BO Location** of any predefined xsd files and click **Next**.



Configure objects window

Result

The objects have been configured for inbound communication.

What to do next

Generate outbound artifacts for the adapter project.

Generating artifacts

Define business object definitions and their related artifacts for use with your connector project.

Before you begin

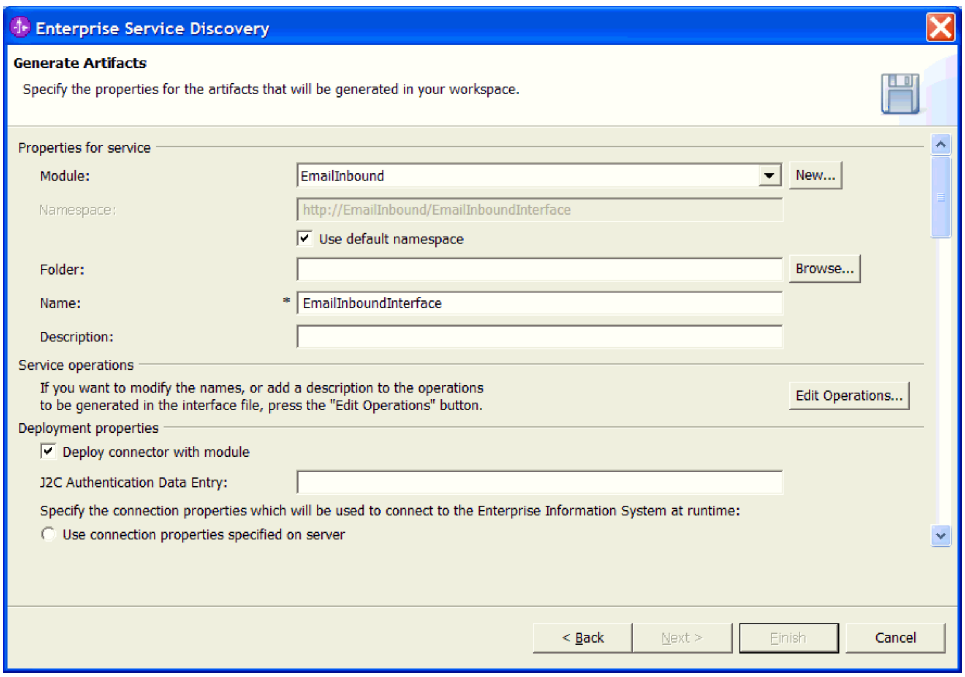
To generate business objects and artifacts for your connector project, you must have already selected business objects to add to the adapter project.

About this task

To generate artifacts for the adapter project, follow this procedure. See "Activation specification properties" on page 122 for more information about these properties.

How to perform this task

1. Next to the **Module** field, click **New**. The New Integration Project window opens.
2. Select **Create a module project** and click **Next**.
3. Name the adapter module and click **Finish**. For inbound communications, consider naming the module EmailInbound. The module is created and a new screen opens to define the managed connection factory properties.
4. **Optional:** To modify the Service operations names or description, click **Edit Operations**.
5. Leave the **Deploy connector with module** check box selected and click **Use discovered connection properties**.
6. **Optional:** Change the default **Host** name and **Port** values to reflect your mail server setup. For more information about event persistence properties, see the activation specification properties in the references section.
7. Change the **Username** and **Password** values to reflect your mail server setup. The UserName and Password must be valid for the e-mail server running on the specified host.
8. **Optional:** Change the **Log file name** and **Trace file name** paths to store the log and trace files in a different location.



Generate Artifacts window

9. Click **Finish** when all of the values have been entered.

Result

The enterprise service discovery wizard will generate the artifacts. The inbound artifacts that are created are visible in the WebSphere Integration Developer Project Explorer under your module.

What to do next

Generate reference bindings for the adapter module.

Generating reference bindings

Create a reference in the assembly editor from the adapter project to an endpoint. The endpoint represents an a generic J2EE component, such as the application server. By wiring the adapter project to the stand-alone reference (represented as a component in the assembly editor), you link the adapter to other server processes.

Before you begin

You must have generated artifacts for your adapter project before creating reference bindings.

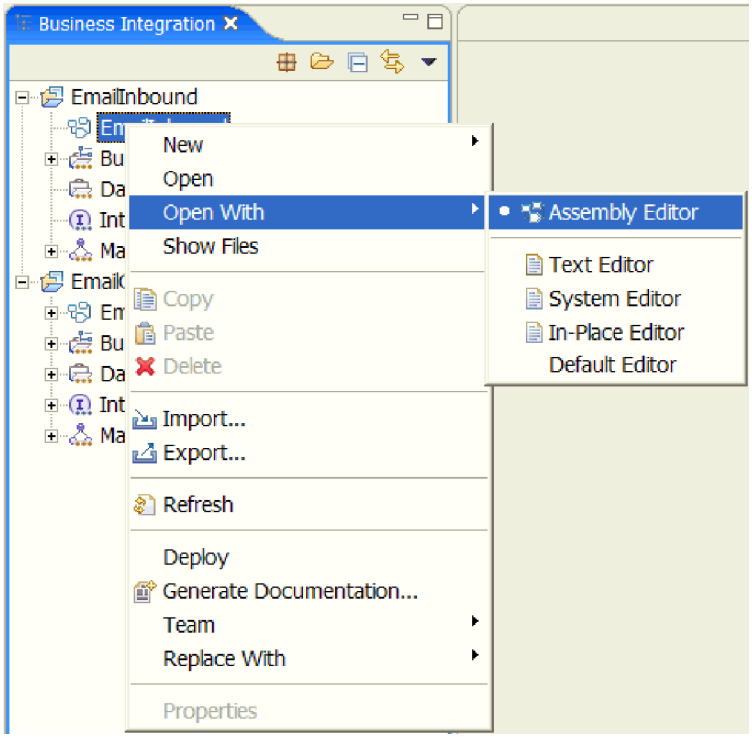
About this task

To generate reference bindings, follow this procedure. For more information on wiring reference bindings, see the WebSphere Integration Developer documentation.

How to perform this task

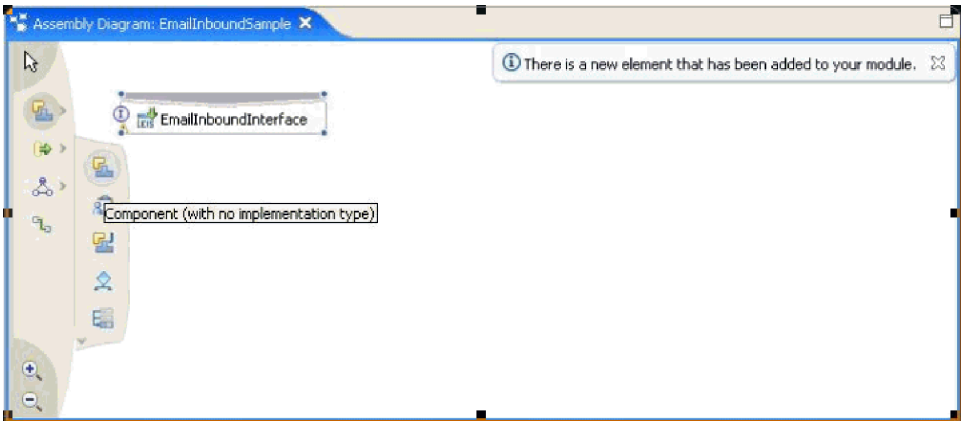
1. **Optional:** Switch to the business integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration Perspective** .

2. Right-click the inbound project and select **Open With → Assembly Editor** .



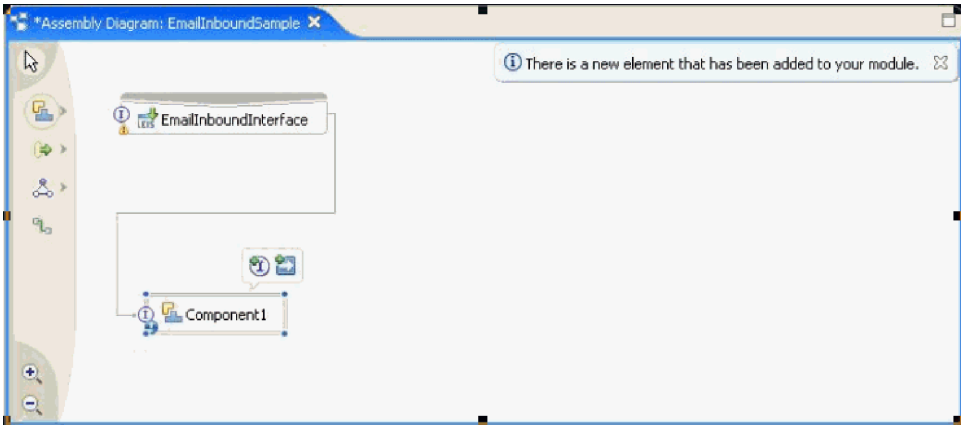
Opening the Assembly Editor

3. Identify and drag the **Component (with no implementation type)** icon into the Assembly Editor. You may need to click once in the assembly editor field to place the component. The component represents a J2EE client, like an application server, that reads inbound requests from the adapter.



Assembly Diagram window with EmailInboundInterface representation

4. Click the wiring icon and select the adapter reference. Then, drag the wire to the component representation.



Assembly Diagram EmailInboundInterface wired to Component 1 representation

5. Click **OK** on any dialog boxes that ask you to save.
6. Right-click the component and select **Generate Implementation → Java**.
7. Select the default package and click **OK**. WebSphere Integration Developer generates the Java implementation for the endpoint component and opens it in a window.
8. Save the generated Component1 Java implementation and the assembly diagram.

Result

A reference binding is created linking the adapter to application server processes.

What to do next

Configure an inbound adapter module or export the outbound adapter module to an EAR file. The EAR file can be deployed to the application server.

Chapter 8. Deploying the module

To deploy the module to the application server, export the adapter project as an enterprise archive (EAR) file, install the module, and add any configuration properties that were not set in the enterprise service discovery wizard.

Exporting the project as an EAR file

Using the enterprise service discovery wizard, export the adapter project that you have created as an EAR file. By creating an EAR file, you capture all of the contents of your adapter project in a format can be easily deployed to the application server.

Before you begin

Before you can export the project as an EAR file, you must have created your business objects and generated reference bindings.

About this task

To export the project as an EAR file, perform the following procedure.

How to perform this task

1. Switch to the J2EE perspective. To do this, select **Window** → **Open Perspective** → **Other** → **J2EE perspective**.
2. Right-click the module and select **Export**.
3. Select EAR file from the Select window.
4. In the EAR Export window, select your project and browse for a destination directory.
5. Click **Finish**.

Result

The adapter project is exported to an EAR file.

What to do next

Install the module in the server administrative console. This deploys the module to application server.

Installing the module

Installing the adapter project is the last step of the deployment process. When you install the adapter project on the server and run it, the adapter, which is embedded as part of the project module, runs as part of the installed application.

Before you begin

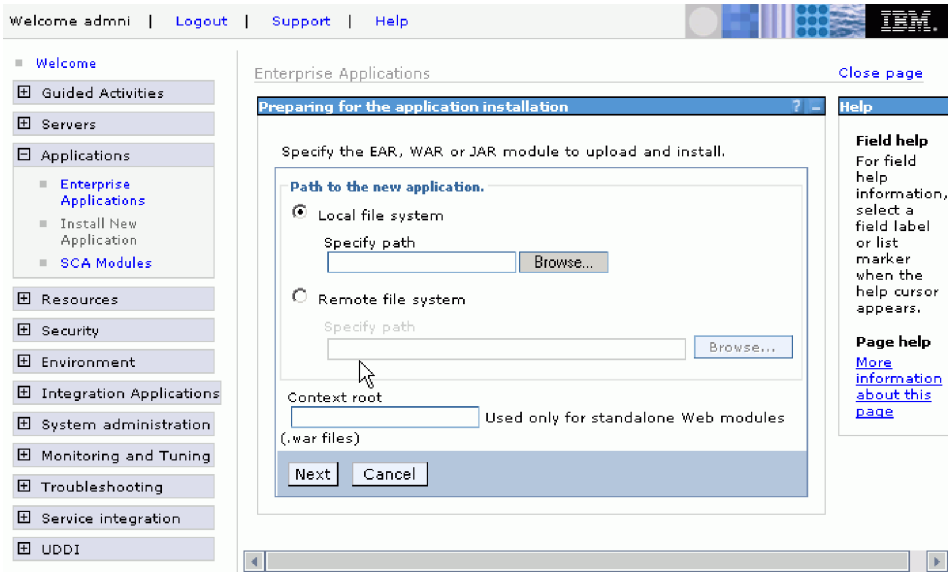
You must have exported your project module as an EAR file before installing the adapter project.

About this task

To install the adapter module, perform the following procedure. For more information on clustering adapter project applications, see http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp.

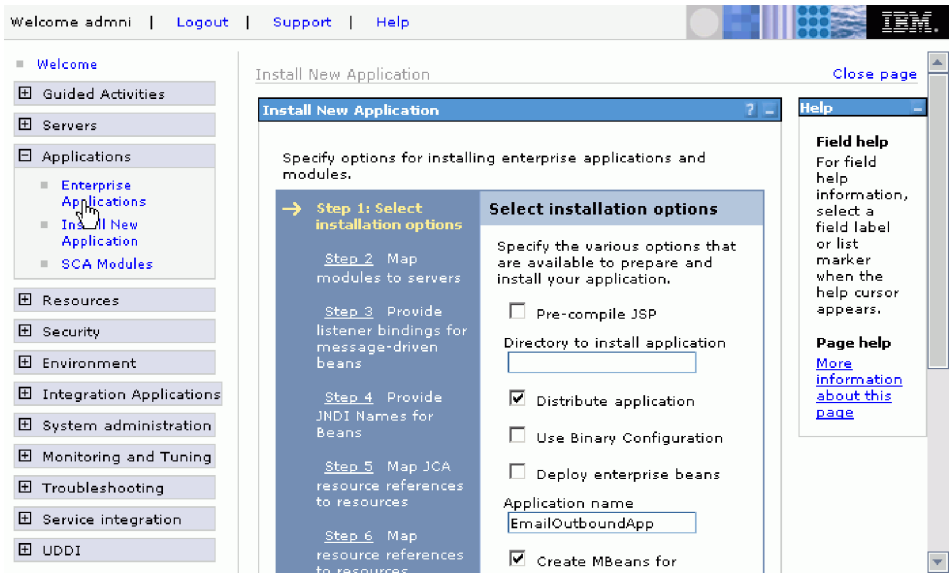
How to perform this task

1. Open the WebSphere Process Server administrative console by right-clicking your server instance and selecting **Run administrative console**.
2. In the administrative console window, click **Applications** → **Install New Applications**.



Preparing for the application installation window

3. Click **Browse** to locate your EAR file and click **Next**.
4. **Optional:** If you are deploying to a clustered environment, click **Next** until you reach Step 2: Mapping modules to servers, then select **Modules** and then the name of the server cluster and click **Apply**. Note: Adapter instances are replicated in a clustered server environment when enableHASupport is set to true. Do not change the value of enableHASupport for single server environments. **Note:** Adapter instances are replicated in a clustered server environment when **enableHASupport** is set to true. Do not change the value of **enableHASupport** for single server environments.
5. Click **Next** until you reach Step 6: Map resource reference to resources.



Install New Application window

6. Select **SCA Auth Alias** from the select authentication data entry list.
7. Select the check box for the module and click **Apply**.
8. Click **Next**. A summary of all of the installation options is displayed.
9. Verify that all options are correct and click **Finish**.
10. Confirm that the application was installed successfully.
11. Click the **Save to Master Configuration** link at the end of the list of installation messages.
12. Click **Save**.

Result

The project is now deployed and the Enterprise Applications window for the deployed application is displayed.

What to do next

If you want to set or reset resource adapter, managed connection factory, activation specification, or data transformation properties, or you would like to cluster adapter project applications, you should do that using the WebSphere Process Server administrative console before configuring troubleshooting tools.

Setting or changing configuration settings from the administrative console

To define a new property or reset a property after you have created your adapter project, use the administrative console.

Setting resource adapter properties

To reset a resource adapter property or to add a property, use the administrative console. Resource adapter properties consist of logging and tracing, bidirectional language support, and activities specific to the adapter, such as the default configuration properties of the adapter. You configure these properties using WebSphere Process Server administrative console.

How to perform this task

See "Resource adapter properties" on page 119 for more information about these properties.

1. Start the WebSphere Process Server administrative console.
2. Select **Application → Enterprise applications**.
3. Double click the deployed EAR module.
4. On the Related Item tab, select **connector module** and then double click CWEM_EMail.rar.
5. Click **Resource adapter → custom properties** to change the resource adapter properties.
6. Make your changes.
7. Save.

Setting managed (J2C) connection factory properties

To reset a managed connection factory property or to add a new value, use the administrative console. Managed connection factory configuration properties are used at run time to create an outbound connection instance with an enterprise information system.

How to perform this task

See "Managed (J2C) connection factory properties" on page 121 for more information about these properties.

1. Start the WebSphere Process Server administrative console.
2. Select **Application → Enterprise applications**.
3. Double click the EAR module you just deployed.
4. On the **Related Item tab**, select **connector module** and then double click CWEM_EMail.rar.
5. Select **Resource adapter → J2C Connection Factories** .
6. Click **EmailOutboundInterface_CF → custom properties** to change the managed connection factory properties.
7. Save.

Setting activation specification properties for the EIS

To reset an activation specification property or to add a new property value, use the administrative console. Activation specification properties hold the inbound event processing configuration information for a message endpoint.

How to perform this task

See "Activation specification properties" on page 122 for more information about these properties.

1. Start the WebSphere Process Server administrative console.
2. Select **Application → Enterprise applications**.
3. Double click the EAR module you just deployed.
4. On the **Related Item tab**, select **connector module** and then double click CWEM_EMail.rar.
5. Select **Resource adapter → EmailInboundInterface_AS**.
6. Click **Custom properties** to edit the activation specification properties.

7. Save.

Chapter 9. Configuring troubleshooting tools

Configure the troubleshooting tools to suit your requirements. Enable logging for the adapter to control the status of event processing. Enable the Common Event Infrastructure to collect diagnostic information about your adapter. Set tracing levels to determine the level of the information captured in the adapter log and trace files. Install IBM Support Assistant to gain quick access to support-related information along with serviceability tools for problem determination for IBM software products.

Enabling tracing with the Common Event Infrastructure (CEI)

Enable tracing and control the level of detail in the adapter trace by configuring the Common Event Infrastructure (CEI).

Before you begin

Before you enable tracing with CEI, complete the following tasks:

- Enable the diagnostic trace service.
- Publish the IBM WebSphere Adapters event definitions file to the CEI catalog before you can set these event definitions.

For instruction on how to do these tasks, refer to the CEI documentation located on the Web site for your server:

- For WebSphere Process Server: http://www.ibm.com/software/integration/wps
- For WebSphere Enterprise Service Bus: http://www.ibm.com/software/integration/wsesb

To enable tracing and control the level of trace detail, use the following procedure.

How to perform this task

1. In the administrative console, click **Troubleshooting**.
2. Click **Logs and Trace**.
3. In the list of servers, click the name of your server.
4. In the General Properties area, click **Change Log Detail Level** and then select **com.ibm.j2ca.*** for the adapter components. There is a subcomponent for each adapter type, as described in the following table.

Adapter	Package Name
WebSphere Adapter for Email	com.ibm.j2ca.email.*
WebSphere Adapter for Flat Files	com.ibm.j2ca.flatfile.*
WebSphere Adapter for FTP	com.ibm.j2ca.ftp.*
WebSphere Adapter for JDBC	com.ibm.j2ca.jdbc.*
WebSphere Adapter for JD Edwards EnterpriseOne	com.ibm.j2ca.jde.*
WebSphere Adapter for SAP Software	com.ibm.j2ca.sap.*
WebSphere Adapter for Siebel Business Applications	com.ibm.j2ca.siebel.*

5. Select the component that matches your adapter. Each adapter component has two subcomponents, one for logging and one for CEI. They are:

- *subcomponent_name*.log.*adapter_ID*
- *subcomponent_name*.cei.*adapter_ID*

For example, com.ibm.j2ca.siebel.cei.*adapter_ID1*. For each instance of a deployed adapter, the system shows a separate ID.

6. Select the CEI adapter ID that you want to enable.
7. From the list, choose the level of business object detail to capture in service component events:
   - **off**. Turn CEI off.
   - **fine**. Turn CEI on but publish none of the business object payload. This corresponds to the event control detail level of Empty in WebSphere Integration Developer.
   - **finer**. Turn CEI on and publish only the payload description for the business object. This corresponds to the event control detail level of Digest in WebSphere Integration Developer .
   - **finest**. Turn CEI on and publish all of the business object payload. This corresponds to the event control detail level of Full in WebSphere Integration Developer.
   - **all**. Same as **finest**.

   For information on what each event content level means (Empty, Digest and Full), and for more information on using the Common Base Event model and the Common Event Infrastructure, refer to the documentation for your process server.

Configuring logging properties

Use the administrative console to enable logging and to set the output properties for a log, including the location, level of detail, and output format of the log.

About this task

Before the adapters can log monitored events, you must specify the service component event points that you want to monitor, what level of detail you require for each event, and format of the output used to publish the events to the logs. Use the administrative console to perform the following tasks:

- Enable or disable a particular event log
- Specify the level of detail in a log
- Specify where log files are stored and how many log files are kept
- Specify the format for log output

  If you set the output for log analyzer format, you can open trace output using the Log Analyzer tool, which is an application included with your process server. This is useful if you are trying to correlate traces from two different server processes, because it allows you to use the merge capability of the Log Analyzer.

For more information about monitoring on a process server, including service components and event points, see the documentation for your process server.

You can change the log configuration statically or dynamically. Static configuration take effect when you start or restart the application server. Dynamic, or runtime, configuration changes apply immediately.

When a log is created, the detail level for that log is set from the configuration data. If no configuration data is available for a particular log name, the level for that log is obtained from the parent of the log. If no configuration data exists for the parent log, the parent of that log is checked, and so on up the tree, until a log with a non-null level value is found. When you change the level of a log, the change is propagated to the children of the log, which recursively propagate the change to their children, as necessary.

To enable logging and set the output properties for a log, use the following procedure.

How to perform this task

1. In the navigation pane of the administrative console, click **Servers** → **Application Servers**.
2. Click the name of the server that you want to work with.
3. Under **Troubleshooting**, click **Logs and trace**.
4. Click **Change Log Detail Levels**.
5. Specify when you want the change to take effect:
   - For a static change to the configuration, click the **Configuration** tab.
   - For a dynamic change to the configuration, click the **Runtime** tab.
6. Select the packages whose logging level you want to modify. The package names for WebSphere Adapters start with **com.ibm.j2ca**:
   - For the adapter base component, select **com.ibm.j2ca.base**.
   - For the adapter base component and all deployed adapters, select **com.ibm.j2ca.base.***.
   - For a specific adapter, select its package name.

Adapter	Package Name
WebSphere Adapter for Email	com.ibm.j2ca.email
WebSphere Adapter for Flat Files	com.ibm.j2ca.flatfile
WebSphere Adapter for FTP	com.ibm.j2ca.ftp
WebSphere Adapter for JDBC	com.ibm.j2ca.jdbc
WebSphere Adapter for JD Edwards EnterpriseOne	com.ibm.j2ca.jde
WebSphere Adapter for SAP Software	com.ibm.j2ca.sap
WebSphere Adapter for Siebel Business Applications	com.ibm.j2ca.siebel

7. Click the package name and select the logging level.

Logging Level	Description
Fatal	The task cannot continue or the component cannot function.
Severe	The task cannot continue, but the component can still function. This logging level also includes conditions that indicate an impending fatal error, that is, situations that strongly suggest that resources are on the verge of being depleted.
Warning	A potential error has occurred or a severe error is impending. This logging level also includes conditions that indicate a progressive failure, for example, the potential leaking of resources.
Audit	A significant event has occurred that affects the server state or resources.

Logging Level	Description
Info	The task is running. This logging level includes general information outlining the overall progress of a task.
Config	The status of a configuration is reported or a configuration change has occurred.
Detail	The subtask is running. This logging level includes general information detailing the progress of a subtask.

8. Click **Apply**.
9. Click **OK**.
10. To have static configuration changes take effect, stop and then restart the process server.

Changing the log and trace file names

By default, log and trace information for all processes and applications on a process server is written to the SystemOut.log and trace.log files, respectively. To keep the adapter log and trace information separate from other processes, use the administrative console to change the file names.

About this task

You can change the log and trace file names at any time after the adapter module has been deployed to an application server.

You can change the log configuration statically or dynamically. Static configuration changes affect applications when you start or restart the application server. Dynamic or run time configuration changes apply immediately.

Log and trace files are in the *install_root*/profiles/*profile_name*/logs/*server_name* folder.

To set or change the log and trace file names, use the following procedure.

How to perform this task

1. In the navigation pane, click **Enterprise Applications**.
2. Click the name of the adapter application. This is the name of the EAR file for the adapter, without the .ear file extension. For example, if the EAR file is named Accounting_OutboundApp.ear, then click **Accounting_OutboundApp**.
3. Click **Connector Modules**.
4. Select the adapter by clicking the name of the RAR file for the adapter. The RAR files are listed in the following table.

Adapter	RAR File Name
WebSphere Adapter for Email	CWYEM_Email.rar
WebSphere Adapter for Flat Files	WYFF_FlatFile.rar
WebSphere Adapter for FTP	CWYFT_FTPFile.rar
WebSphere Adapter for JDBC	CWYBC_JDBC.rar
WebSphere Adapter for JD Edwards EnterpriseOne	CWYED_JDE.rar
WebSphere Adapter for SAP Applications	CWYAP_SAPAdapter.rar CWYAP_SAPAdapterTX.rar

Adapter	RAR File Name
WebSphere Adapter for Siebel Business Applications	CWYEM_Siebel.rar

5. Click the name of the resource adapter.
6. In the Custom Properties area, specify the file names:
   - To change the log file name, type the name in the **Value** field for **logFilename**. By default, this log is in the SystemOut.log file.
   - To change the trace file name, type the name in the **Value** field for **traceFilename**. By default, this log is in the trace.log file.
7. To have static configuration changes take effect, stop and then restart the process server.

Installing or upgrading IBM Support Assistant

IBM Support Assistant (ISA) is a free, local software serviceability workbench that helps you resolve questions and problems with IBM software products. Install plug-ins for the products you have installed. It provides quick access to support-related information along with serviceability tools for problem determination. Installing and upgrading it is simple and straightforward.

About this task

IBM Support Assistant provides the following services:

- Symptom-based data collection
- Access to IBM support information, IBM newsgroups, and other resources through a federated search interface (one search, multiple resources)
- Easy access to IBM educational materials
- Easy access to IBM product home pages, product support pages, and product forums or newsgroups through convenient links
- A tools framework and update manager to easily update and install ISA plug-ins and tools
- Fast resolution of problem management records through electronic submission of critical system data to IBM

You can install and run both version 2 and version 3 of IBM Support Assistant on a single computer, to get support for a broad range of IBM solutions.

To install and upgrade IBM Support Assistant, use the following procedure.

How to perform this task

1. Go to the IBM Support Assistant Web page at:
   http://www.ibm.com/software/support/isa/
2. Follow the directions on the Web page to download ISA version 3.0, and then to extract, install, and use the tool.
3. Start ISA.
4. Open the **Updater** component.
5. On the **Upgrades** tab, upgrade ISA to version 3.0.1 or higher.
6. On the **New Products and Tools** tab, install the plug-ins for your adapter. Select the plug-in for your adapter from the list for the WebSphere brand. There is an optional language pack plug-in for each adapter, which enables you to see adapter-specific information in languages other than English.

Chapter 10. Administering the adapter

Use the administrative console of the server to start, stop, and troubleshoot the adapter.

Starting the adapter

To start an adapter that has a status of Stopped, use the administrative console. By default, an adapter starts automatically when the server starts.

Before you begin

The administrative console of the server must be running in order to complete this task.

To start the adapter, use the following procedure.

How to perform this task

1. On the Enterprise Applications page, click **Applications** → **Enterprise Applications**.
2. Select the check box of the adapter that you want to start.
3. Click **Start**.

Result

The status of the adapter changes to Started and a message stating that the adapter started displays at the top the page.

Use the administrative console of the server to stop the adapter.

Stopping the adapter

Use the administrative console of the server to stop an adapter.

Before you begin

The administrative console of the server must be running in order to complete this task.

To stop the adapter, use the following procedure.

How to perform this task

1. On the Enterprise Applications page, click **Applications** → **Enterprise Applications**.
2. Clear the check box of the adapter you want to stop.
3. Click **Stop**.

Result

The status of the adapter changes to Stopped and a message stating that the adapter stopped displays at the top the page.

Use the administrative console of the server to troubleshoot the adapter.

Troubleshooting and support

Common troubleshooting techniques and self-help information help you identify and solve problems quickly. If necessary, follow the procedures for contacting IBM Software Support.

Exception: XAResourceNotAvailableException

When the process server log contains repeated reports of the com.ibm.ws.Transaction.XAResourceNotAvailableException exception, remove transaction logs to correct the problem.

Symptom:

When the adapter starts, the following exception is repeatedly logged in the process server log file:

com.ibm.ws.Transaction.XAResourceNotAvailableException

Problem:

A resource was removed while the process server was committing or rolling back a transaction for that resource. When the adapter starts, it tries to recover the transaction but cannot because the resource was removed.

Solution:

To correct this problem, use the following procedure:

1. Stop the process server.
2. Delete the transaction log file that contains the transaction. Use the information in the exception trace to identify the transaction. This prevents the server from trying to recover those transactions.

   Note: In a test or development environment, you can generally delete all of the transaction logs. In WebSphere Integration Developer, delete the files and subdirectories of the transaction log directory, *server_install_directory*\profiles*profile_name*\tranlog.

   In a production environment, delete only the transactions that represent events that you do not need to process. One way to do this is to reinstall the adapter, pointing it to the original event database used, and deleting only the transactions you do not need. Another approach is to delete the transactions from either the log1 or log2 file in the following directory:

   server_install_directory\profiles*profile_name*\tranlog*node_name*\wps*server_name*\transaction\tranlog
3. Start the process server.

Self help resources

Use the self help resources of IBM Software Support to get the most current support information, to obtain technical documentation, to download support tools and fixes, and prevent problems with WebSphere Adapter for Email. The self help resources also help you diagnose problems with the adapter and contact IBM Software Support.

The software support Web site for WebSphere Adapters at http://www.ibm.com/software/integration/wbiadapters/supp provides the following resources:

- Flashes (alerts from technical support)
- Technotes

  You can get a list of technotes for WebSphere Adapters at http://www.ibm.com/support/search.wss?rs=695&tc=SSMKUK
- Authorized program analysis reports (APARs)
- Technical information including the product information center, manuals, IBM Redbooks™, and whitepapers.
- Educational offerings
- *IBM Software Support Handbook*

Register at the site to use My Support to create a customized support page for your use.

Contacting IBM Software Support

IBM Software Support provides support for WebSphere Adapters either online or by phone. Gathering information about the problem before you contact IBM Software Support can dramatically increase support responsiveness.

Before you begin

If you think your problem is defect-related, IBM Software Support provides assistance. Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have:

- For IBM distributed software products (including, but not limited to, Tivoli®, Lotus®, and Rational® products, as well as DB2® and WebSphere products that run on Windows, Linux®, or UNIX® operating systems), you must be enrolled in Passport Advantage®. You can enroll in one of the following ways:

  Online

  Go to the Passport Advantage Web page (http://www-306.ibm.com/software/support/pa.html), and click **How to Enroll**.

  By phone

  For the phone number to call in your country, go to the contacts page of the IBM Software Support Handbook on the Web (http://techsupport.services.ibm.com/guides/contacts.html), and click the name of your geographic region.
- For IBM eServer™ software products (including, but not limited to, DB2 and WebSphere products that run in zSeries®, pSeries®, and iSeries™ environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM Technical Support Advantage Web page (http://www-03.ibm.com/servers/eserver/techsupport.html).

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States or, from other countries, go to the contacts page of the IBM Software Support Handbook on the Web

(http://techsupport.services.ibm.com/guides/contacts.html), and click the name of your geographic region for phone numbers of people who provide support for your location.

About this task

The IBM Software Support Handbook contains detailed information about the service and support of your IBM products. Read the handbook at http://techsupport.services.ibm.com/guides/handbook.html.

To contact IBM Software Support, use the following procedure.

How to perform this task

1. Describe your problem and gather background information. When explaining a problem to a support specialist, be as specific as possible. Include all relevant background information so that the specialists can help you solve the problem efficiently. To save time, know the answers to these questions:
   - What software versions were you running when the problem occurred? Include the version of the operating system as well as related products.
   - Has the problem happened before, or is this an isolated problem?
   - What steps led to the failure?
   - Can the problem be recreated? If so, what steps led to the failure?
   - Have any changes been made to the system such as to the hardware, operating system, networking software, and so on?
   - Are you currently using a workaround for this problem? If so, be prepared to explain it when you report the problem.
   - Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
2. Determine the business impact of your problem. When you report a problem, you will be asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem you are reporting. Use the criteria described in the following table.

   Table 4. Severity criteria for problem reporting

Severity	Description
1	**Critical business impact**: You are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.
2	**Significant business impact**: The program is usable but is severely limited.
3	**Some business impact**: The program is usable with less significant features (not critical to operations) unavailable.
4	**Minimal business impact**: The problem causes little impact on operations, or a reasonable circumvention to the problem has been implemented.

3. Submit your problem to IBM Software Support. You can submit your problem in the following ways:
   - **Online**. Go to the Submit and track problems page on the IBM Software Support site http://www.ibm.com/software/support/probsub.html Enter your information into the appropriate problem submission tool.

- **By phone**. For the phone number to call in your country, go to the contacts page of the IBM Software Support Handbook on the Web (http://techsupport.services.ibm.com/guides/contacts.html), and click the name of your geographic region.

Result

If the problem you submit is for an unreported software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail and tracks its resolution.

What to do next

Whenever possible, IBM Software Support provides a workaround for you to implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the product support Web pages daily, so that other users who experience the same problem can benefit from the same resolution.

Chapter 11. Quick start tutorials

To gain practical knowledge in setting up and deploying the adapter, complete one or more of the tutorials. Everything you need to complete each tutorial is contained in the tutorial. If you have performed the prerequisite tasks (such as installing the adapter), you can complete each tutorial in under an hour.

Introduction

Each tutorial provides a complete set of instructions for configuring the adapter so that it can be used by an application server to send requests to a mail server or by a mail server to send requests to an application server.

In the tutorials, you use WebSphere Integration Developer (and its enterprise service discovery wizard) to configure the adapter, connect to a mail server, and retrieve information about a service on the mail server. Enterprise service discovery then creates the business objects and interface information needed to interact with the service. The business objects and interface information, along with the adapter, are built into a deployable module.

Learning objectives for the outbound scenarios

After completing the outbound tutorials, you should be able to perform the following tasks:

- The outbound request processing capabilities of the Email adapter
- Data transformation capabilities for known mime types
- Passing through of content without specific mime types
- Function of the SMTP protocol
- Support for mail attachments

Time required for the outbound scenarios

If the mail server and mail client is set up and configured, the tutorials should take the following amounts of time:

- Sending outbound data using the sample EAR file– 30 minutes
- Creating and exporting an outbound EAR file– 45 minutes

Learning objectives for the inbound scenario

After completing the inbound tutorials, you should be able to perform the following tasks:

- The inbound event processing capabilities of the Email adapter
- Data transformation capabilities for known mime types
- Data transformation of attachments
- Function of IMAP and POP3 protocols
- Archival of events

Time required for the inbound scenario

If the mail server and mail client is set up and configured, the tutorials should take the following amounts of time:

- Retrieving inbound data using the sample EAR file– 30 minutes
- Creating and exporting an inbound EAR file– 45 minutes

Audience

The tutorials are intended for the integration developer who will be configuring Adapter for Email software for deployment on WebSphere Process Server or WebSphere Enterprise Service Bus.

Prerequisites

Before you begin the tutorials, make sure you have performed the following tasks:

- Install all prerequisite software
- Install Adapter for Email software
- Confirm that the Java mail and Activation jar files are in the adapter runtime class path.

  Note: These files are in the RAR file, but you may need to update them if you are not using the most current version of WebSphere Integration Developer.

Also, make sure you have all the information (such as user ID, password, host, and port) needed to access the mail server.

Many of the tutorials included in this section list importing the RAR file as one of the first steps. It is not necessary to import this file multiple times. You can use the same RAR file for all of the scenarios. Or, if you prefer, you can create a new workspace for each tutorial and import the RAR file each time.

Accessing the tutorial files

Accessing the tutorial files is an optional task. These steps have been documented for advanced users who wish to explore the internal contents of the samples. First-time users or those not interested in the files themselves may skip this section. To access the tutorial files, run the installer and extract the ear files located in the Samples folder.

1. Run the adapter installer. If you have not already installed the adapter, do so now. You may install the adapter into any directory. CWYEM_EMail.RAR and a Samples folder containing an ear file for each sample scenario will be placed in your directory.
2. Extract the sample files:
   a. Navigate to the Samples folder and unzip each of the EAR files (EmailInboundSampleApp.ear and EmailOutboundSampleApp.ear) into separate subfolders. Consider using meaningful names to identify which ear file contents correspond to each sample scenario. For example, call one folder InboundSample.
   b. Double-click each EAR file and extract their contents to a separate Samples subfolders for each EAR file. The files that are extracted by each EAR file are precise examples of the artifacts the user creates when using the

enterprise service discovery wizard in each scenario. In addition, they can be used as reference to verify what files created by enterprise metadata discovery should look like.

Result

When each EAR file has been extracted, the following files should be present in your directory.

Files in EmailInboundSampleApp.ear

File/Path	File description
CWYEM_EMail.rar	The RAR file contains the binaries of the Email RA.
EmailInboundSample.jar	The jar has the sca files (sca.module, sca.modulex), xsds and other artifacts generated by the Email EMD.
EmailInboundSampleEJB.jar	The jar file contains a generated EJB implementation of the endpoint component associated with the inbound service
EmailInboundSampleEJBClient.jar	The jar file contains the generated EJB Client implementation with the above EJB.
META-INF folder	Contains ibm-application-ext-pme.xmi, application.xml, ibmconfig folder, along with the manifest file.
EmailInboundSample.jar\xsds	Will contain all the XSDs generated by the Email EMD for the module. This would include the files – Email.xsd, EmailBG.xsd, Header.xsd, MailAttachment.xsd, UnstructuredContent.xsd, WbiAddress.xsd, WbiPhone.xsd, WbiAddressWrapper.xsd, WbiAddressWrapperBG.xsd
EmailInboundSample.jar\ EmailInboundInterface.wsdl	WSDL interface for the inbound service
EmailInboundSample.jar\ EmailInboundInterface.export	Export file for the inbound service
EmailInboundSample.jar\ Endpoint.component	The implementation of this sample component endpoint will be available at EmailEMD.jar\sca\component\java\impl

Files in EmailOutboundSampleApp.ear

File/Path	File description
CWYEM_EMail.rar	The RAR file contains the binaries of the Email RA.
EmailOutboundSample.jar	The jar has the sca files (sca.module, sca.modulex), xsds and other artifacts generated by the Email EMD.
EmailOutboundSampleEJB.jar	The jar file contains a generated EJB implementation of the endpoint component associated with the outbound service
EmailOutboundSampleEJBClient.jar	The jar file contains the generated EJB Client implementation with the above EJB.

File/Path	File description
MyWebProject.war	Contains emailwebclient.jsp, with other web-module artifacts like the services.properties file. emailwebclient.jsp will trigger the outbound request to the Email adapter.
META-INF folder	Contains ibm-application-ext-pme.xmi, application.xml, ibmconfig folder, along with the manifest file.
EmailOutboundSample.jar\xsds	Will contain all the XSDs generated by the Email EMD for the module. This would include the files – Email.xsd, EmailBG.xsd, Header.xsd, MailAttachment.xsd, UnstructuredContent.xsd, WbiAddress.xsd, WbiPhone.xsd, WbiAddressWrapper.xsd, WbiAddressWrapperBG.xsd
EmailOutboundSample.jar\ EmailOutboundInterface.wsdl	WSDL interface for the outbound service
EmailOutboundSample.jar\ EmailOutboundInterface.import	Import file for the outbound service

Preparing to run the outbound scenarios

You must have a configured mail server and mail client to run the outbound tutorials. You can use any mail client and mail server you choose, but the mail server must supports IMAP/POP3 protocols for inbound operations and SMTP for outbound operations.

Tutorial 1: Sending outbound data using the sample EAR file

In this tutorial you will use the EAR file supplied with the samples material to see how the adapter processes outbound requests, how data is transformed for known MIME types, and how data is passed through the system. You will also learn how about SMTP protocol functions and mail-attachment support.

Running the sample EAR in WebSphere Integration Developer

Since the EAR file is shipped as the sample for this tutorial, you do not need to run the enterprise service discovery wizard to generate your business object artifacts. In this scenario, you will deploy the sample EAR file to the test server, send requests to the adapter, and then verify the results.

Before you begin

Before you deploy the sample EAR file you must configure the mail server and mail client.

How to perform this task

1. Open WebSphere Integration Developer and switch to the J2EE Perspective.
2. **Optional:** On the Servers tab, right-click the server and select **Start** if the server is not already running.
3. Select **File → Import**.
4. Select **EAR** file from the menu and then click **Next**.

5. For the EAR file field, click **Browse** and navigate to the Samples folder on your system.
6. Select EmailOutboundSampleApp.ear and click **Open**. This will populate the EAR file field on the Import dialog.
7. **Finish** to import the EAR file. The EAR file is now available under Enterprise Applications in the Project Explorer window.
8. Right-click the server instance and select **Add and remove objects**.
9. In the left pane of the Add and Remove Projects window, select EmailOutboundSampleApp and click **Add**. EmailOutboundSampleApp will move to the Configured Projects panel.
10. Select **Finish**.

Result

The EAR file is deployed to WebSphere Process Server test server.

Configuring J2C connection factory properties for mail server connectivity

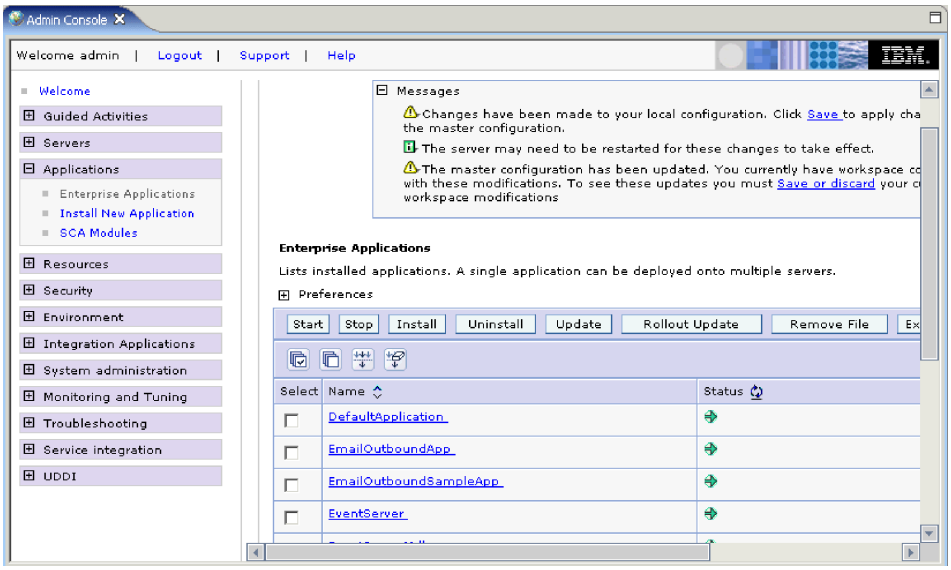
For the adapter to access the mail server, you must change the username and password values in the J2C connection properties through the WebSphere Process Server administrative console. In addition, if you are using a remote mail server, you must also change the host and port values.

About this task

If you have not installed the mail server on the same system as WebSphere Process Server or Enterprise Service Bus, set the host and port number properties in the managed connection factory custom properties from the WebSphere Process Server Administrative console as noted in the following steps. Regardless of where your mail server is installed, you must configure two users on your mail server for the adapter project. To do this, follow the steps listed below.

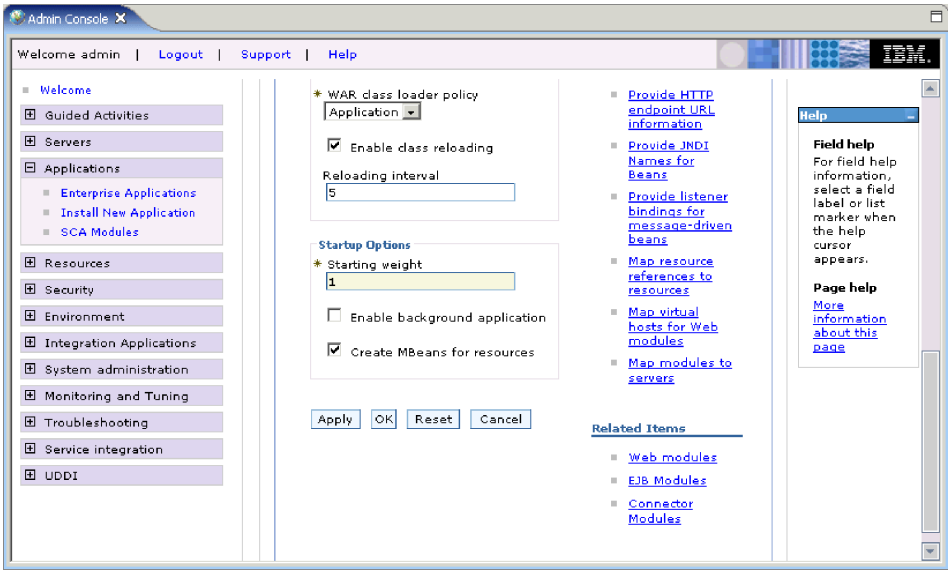
How to perform this task

1. In WebSphere Integration Developer, switch to the J2EE perspective by selecting **Window → Open perspective → Other → J2EE Perspective.**
2. Confirm that the server is started. If it is not, right-click the server instance and start the server.
3. Open the server administrative console by right clicking the server instance and selecting **Run administrative console**.
4. In the admin console, expand **Applications** and click **Enterprise Applications**.



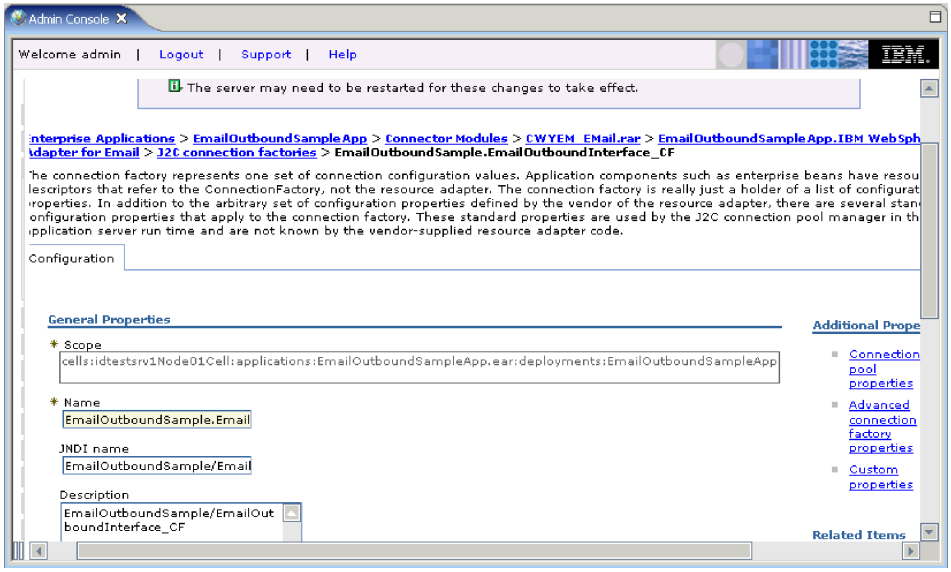
Enterprise Applications window

5. Click **EmailOutboundSampleApp** and then scroll to the very bottom of the screen and select **Connector Modules**.

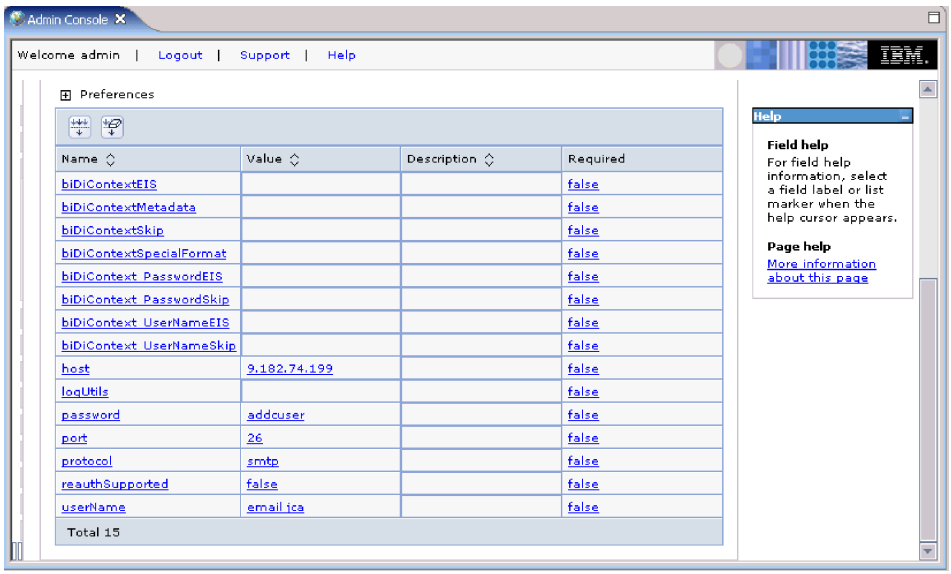


Select Connector modules

6. Select **CWYEM_Email.rar** and then, under Additional Properties, select **Resource Adapter**.
7. Under Additional Properties, select **J2C connection factories**.
8. Select **EmailOutboundSample.EmailOutboundInterface_CF**.
9. Under Additional Properties, select **Custom Properties**.



Custom properties window The Custom Properties screen appears. From this screen, you can select and then change any of the managed connection factories property values.



Click to change property values

10. Click **userName** to open the userName property dialog and change the mail server name. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
11. **Optional:** Click **password** to open the password property dialog and change the mail server password. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
12. **Optional:** If you are using a remote mail server, click **host** to open the host property dialog and change the mail server host. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
13. **Optional:** If you are using a remote mail server, you may need to change your port. If so, click **port** to open the port property dialog and change the mail server port. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.

14. Save your configuration changes.

Result

The adapter has all of the information it will need to connect with the mail server.

Sending requests to the adapter

Once the application is started, you can use the emailwebclient.jsp file provided with the sample to send requests to the Email adapter. This file will simulate a J2EE client invoking an outbound request so that you may verify the that the adapter works correctly.

Before you begin

Before you can run the JSP file, you must have deployed and started the sample EAR file to the WebSphere Process Server test server.

How to perform this task

1. Open a web browser such as Internet Explorer or Firefox.
2. Type the following URL in the browser window: http://localhost:9080\ MyWebProject\emailwebclient.jsp Note that the port number in the example URL may be different than in your setup. Make sure that you specify the port and IP numbers that you specified in your server profile. An Email Resource Adapter Service Client will open.
3. Supply the From, To, CC, BCC, and Subject information you wish to use. The ″To″ field is a required field. All mail addresses used with the jsp file must be in ″mailid@mydomain.com″ format.
4. In the Pass through Content field, enter anything you like. For example, type ″Hi! Look at this wonderful outbound mail!″
5. **Optional:** For the File References field, specify the path to any local files that you would like to attach to the test mail. For example, when C:\EmailRA\Testfile.txt is typed in the File References field (and it exists), Testfile.txt will be attached. If it is not found, a .notfound message will be included in the e-mail.
6. When you have finished specifying the e-mail fields, click **Submit**.

Result

Once the **Submit** button has been clicked, the request will be created. The Email adapter service will be invoked with this request. When a response is created and a mail is sent to the user specified in the ″To″ field, the recipient will receive a message on the same JSP window as follows:

```
Locating the EmailOutboundInterface service...
Invoked the EmailOutboundInterface Successfully.
Response mail sent out successfully
Check out the mail server for results!
```

Verifying the test results

To determine whether your test was successful, use the Email client to verify that your mail was sent by the mail server. You can also check if your test was successful using the JSP client to examine the System.out log or trace.log files under the log/server1 directory.

Verifying test results using the Email adapter client

Check whether your test execution was successful by examining an e-mail sent by the mail server.

If the content is visible as direct content and the attachments (if any) are present, the e-mail delivery was successful.

Result

If your test was successful, you are finished with this tutorial.

What to do next

If your test was unsuccessful, troubleshoot the tutorial.

Troubleshooting the tutorial

If WebSphere Integration Developer generates any errors while running the sample scenario, deploy the provided EAR file directly onto WebSphere Process Server using the administrative console.

Exporting the project into an EAR file

Before you can redeploy the project directly through the WebSphere Process Server Administrative Console, you must re-create your EAR file. This is highly recommended in order to eliminate the possibility of any build errors as a cause of the deployment problem.

1. Confirm that there are no errors by building the project.
2. In WebSphere Integration Developer, export the project into an EAR file.
   a. Right-click the project and select **Export → EAR file**.
   b. In the EAR Export window, select the EAR project and provide the absolute path (including the EAR file name) for the Destination property.
   c. Select the Export Source files, Overwrite existing file, and Include project build paths and meta-data files check boxes.

Result

The EAR file is saved in the destination folder.

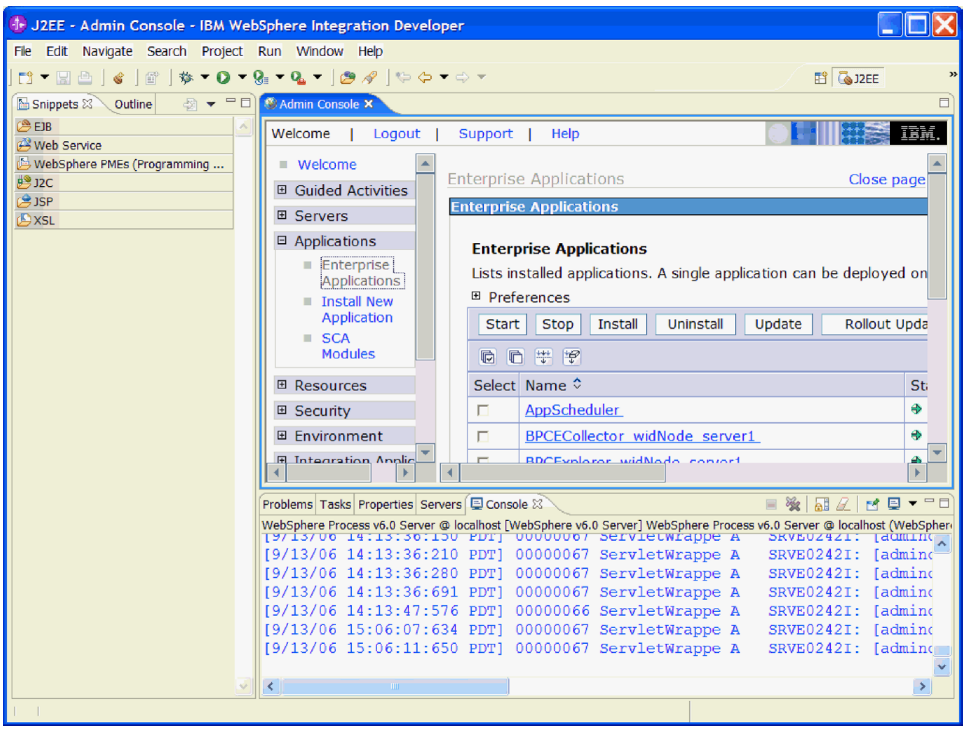
What to do next

If you have successfully exported the project to an EAR file, redeploy the module directly to WebSphere Process Server using the administrative console.

Redeploying the module in the administrative console

After you have recreated your EAR file, redeploy it directly to the test server using the WebSphere Process Server administrative console.

1. Click the Servers tab and verify that the server is started.
2. **Optional:** If it is not already started, right-click on the server instance and start the server.
3. Right-click the server and select **Run adminstrative console**.
4. Either leave the field blank or type Admin in the User ID field and then click **login**. The User ID is used to identify who has logged into the server and is not required.
5. Expand **Applications → Enterprise Applications** and then click **Install**.



Re-installing the EAR file

6. **Browse** for the EAR file and click **Next**.
7. Click **Next** for the Preparing the application for installation window.
8. On the Install New Application window, click **Step 10: Summary** and then **Finish**. A message will confirm when the installation of the application has completed.
9. Click **Save** in the Enterprise Applications window. The application is now installed and saved successfully.

Result

The module is redeployed to the test server.

Tutorial 2: Creating and exporting an outbound EAR file

In this tutorial you will create your own EAR file and deploy it to the test server. You will see how the adapter processes outbound requests, how data is transformed for known MIME types, and how data is passed through the system. You will also learn about SMTP protocol functions and mail-attachment support.

Creating the adapter project

To begin the process of creating a module to communicate with an Email service, you create an adapter project. The adapter project contains the adapter itself plus other related artifacts. You create the project by importing the RAR file, which was copied to your local file system during installation, into WebSphere Integration Developer.

About this task

To create an adapter project, use the following procedure.

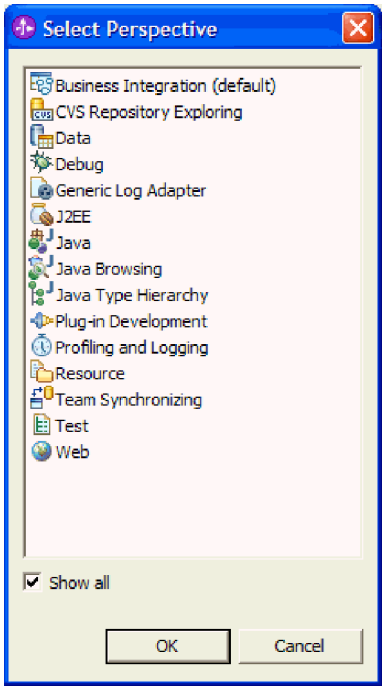
How to perform this task

1. Launch WebSphere Integration Developer by clicking **Start** → **Programs** → **IBM WebSphere** → **Integration Developer 6.0** → **WebSphere Integration Developer 6.0**.
2. If you are prompted to specify a workspace, accept the default value.

   The workspace is a directory where WebSphere Integration Developer stores your project.
3. When the WebSphere Integration Developer window is displayed, close the Welcome page.
4. Switch to the J2EE perspective:
   a. Click **Window** → **Open Perspective** → **Other**.
   b. Click **J2EE**.

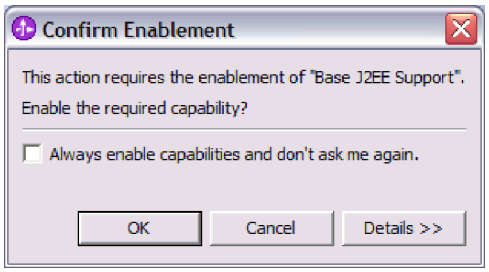
      If **J2EE** is not displayed, select the **Show all** check box, click **J2EE**, and click **OK**.

Figure 7. Selecting J2EE from the Select Perspective list



   c. If you see the Confirm Enablement window, select **Always enable capabilities and don't ask me again**.

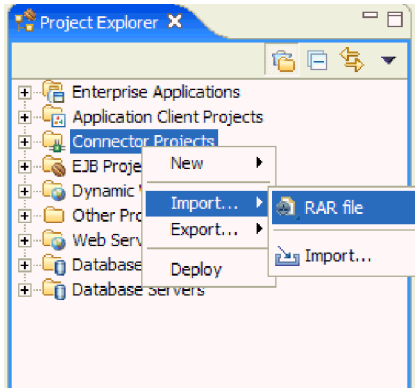
Figure 8. The Confirm Enablement window



d. Click **OK**.

5. Import the RAR file by right-clicking **Connector Projects** and clicking **Import → RAR file**.

Figure 9. Importing the RAR file

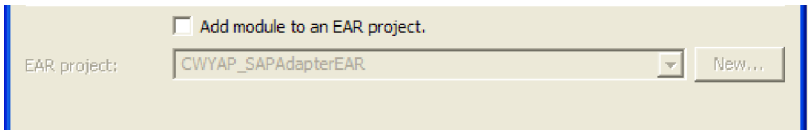


6. Find the RAR file on your local file system by clicking **Browse** and navigating to the directory in which Adapter for Email was installed.
7. Accept the default setting (**CWYEM_Email.rar**) for **Connector project**.

   The connector project has the same name as the RAR file. If project with same name already exists, a number is appended to the name (e.g. CWYEM_EMail1)
8. Accept the default value in the **Target server** field.

   The default value is the test environment for WebSphere Process Server, which is installed as part of WebSphere Integration Developer.
9. Clear the **Add module to an EAR project** check box.

Figure 10. Clearing the Add module to an EAR project check box



   Notice that the EAR project field becomes unavailable after you remove the check mark.
10. Click **Finish**.

Result

The RAR file is imported and a connector project is created. The connector project is visible in the Project Explorer window under Connector Projects.

Configuring the adapter for outbound processing

Use the enterprise service discovery wizard to select business objects and services and to generate artifacts. You can run this tutorial in pass-through or data transformation mode by defining the configuration settings differently.

Setting connection properties for enterprise service discovery

Use enterprise service discovery to determine which services are available to the adapter and to configure the adapter's mail server connection settings.

Before you begin

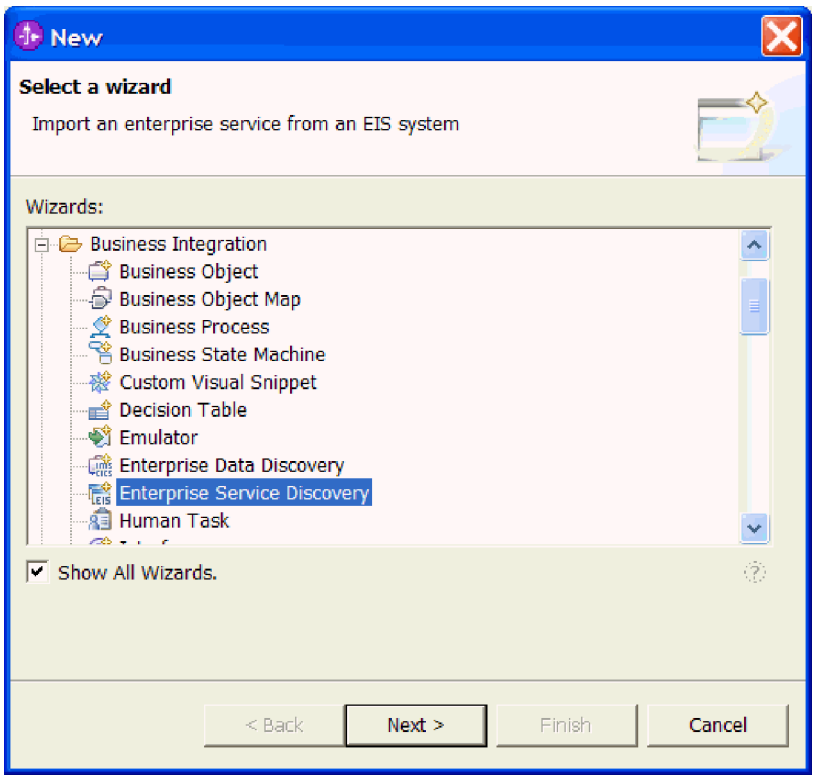
Before you can set the connection properties in this section, you must have created your adapter project.

About this task

To set enterprise service discovery connection properties, follow this procedure.

How to perform this task

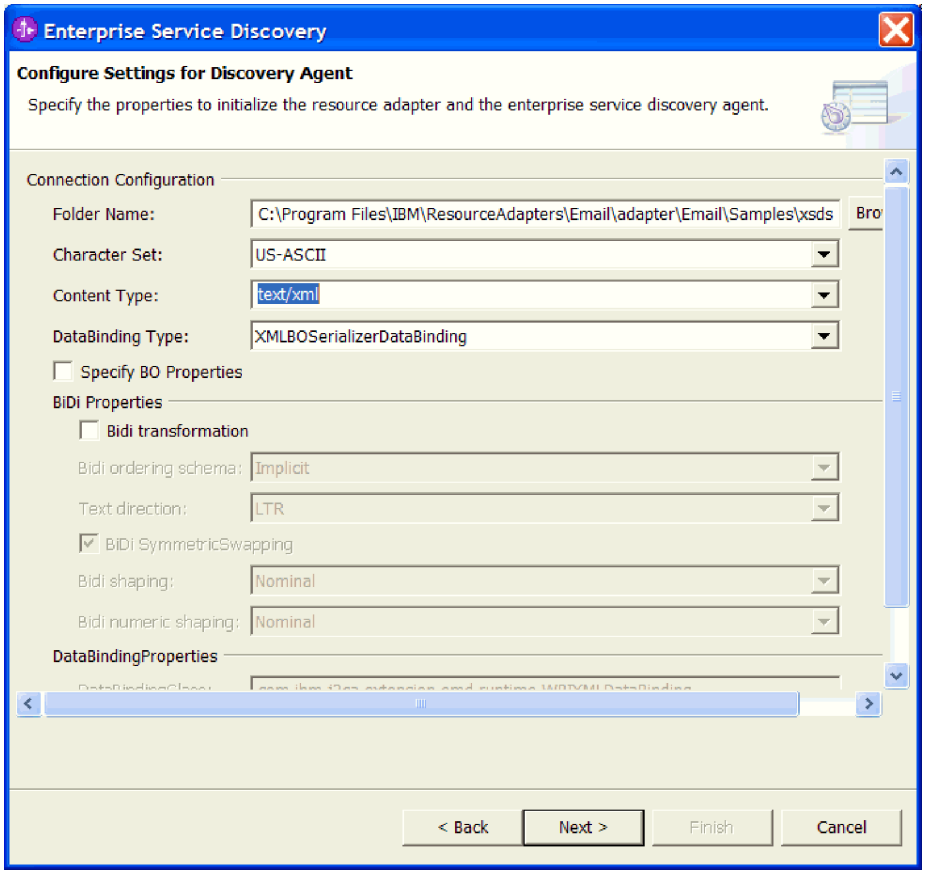
1. Change to the Business Integration perspective by selecting **Window → Open perspective → Other → Business Integration**.
2. Select **File → New → Enterprise Service Discovery**.
3. **Optional:** If Enterprise Service Discovery is not an option, complete the following steps.
   a. Select **File → New → Other**.
   b. Select the **Show all wizards check box** and then expand **Business Integration**.
   c. Select **Enterprise Service Discovery** and then click **Next**.



Select a wizard window

4. Select the Resource Adapter and click **Next**.
5. Make the following selections for the Configure settings for Discovery Agent window:
   a. Browse for the xsd files provided with the sample. The default installation location for the sample xsd file zip is **Program Files** → **IBM** → **ResourceAdapters** → **Email** → **adapter** → **Email** → **Samples** → **XSDs**.

      Note: For a pass-through scenario (without data transformation), leave this field blank.
   b. Select US-ASCII for the **Character Set**.
   c. Select text/xml for the**Content Type** format. When the content type is selected, the corresponding data binding types fields are populated.
   d. Leave the **BiDi transformation** and **Specify BO Properties** check boxes unchecked.
   e. **Optional:** To change the log and trace file directory, click the **Show Advanced** button and browse for a new directory.



Configure settings for Discovery Agent window

f. Select **Next**.

Result

Using these properties, the enterprise service discovery wizard will initialize the adapter and the enterprise service discovery agent.

What to do next

Select the business objects and services to associate with the adapter module.

Selecting business objects and services

Run a query to introspect what services are available to the adapter. The enterprise service discovery wizard displays a metadata tree of all the business objects discovered by the query from which you will select which business object to add to the module.

Before you begin

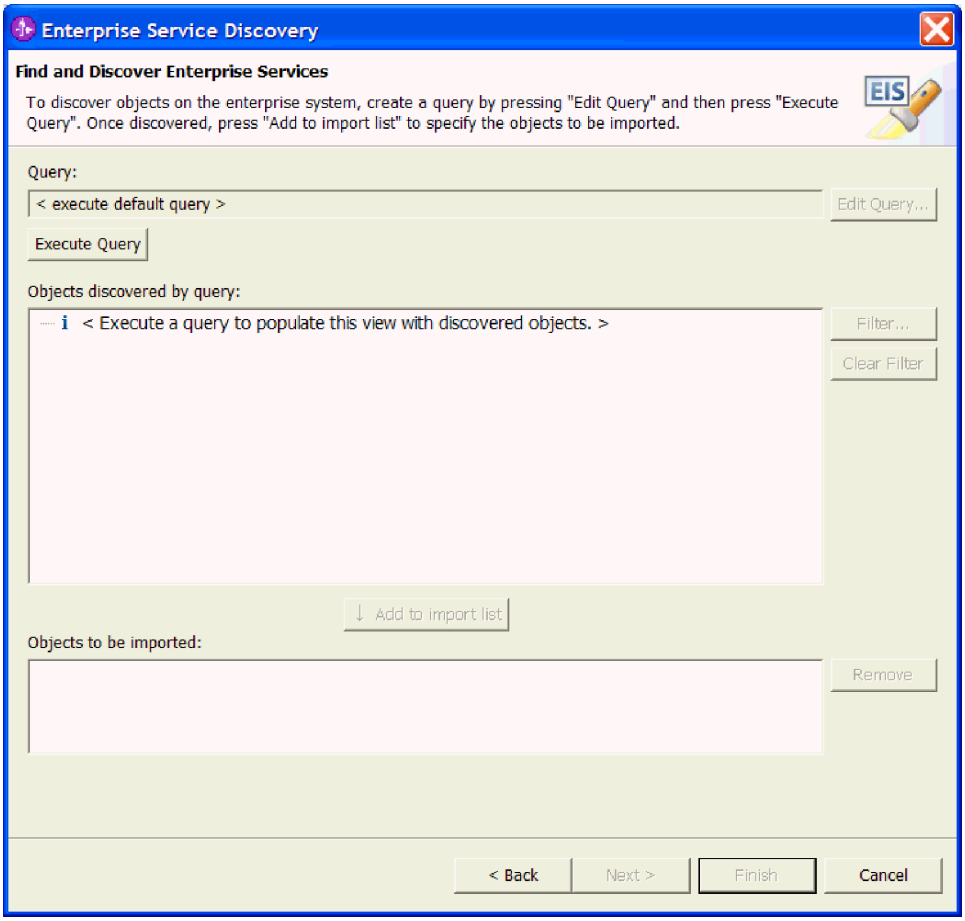
You must have specified the adapter configuration properties for the discovery service before you can complete the steps below.

About this task

To select business objects and services to be used with the adapter, follow this procedure.

How to perform this task

1. Click **Execute Query**. Any business objects found in the folder defined on the last screen are displayed under Objects discovered by query.
2. Select the business objects and click **Add to import list**.



Find and Discover Enterprise Services window

3. The business objects are added to Objects to be imported. Click **Next**.

Result

Using these properties, the discovery service prepares a metadata tree that will be displayed for object selection and navigation in the following steps.

What to do next

Configure the selected business objects.

Configuring the selected objects

Once you have added business objects to the module, configure them for outbound operations.

Before you begin

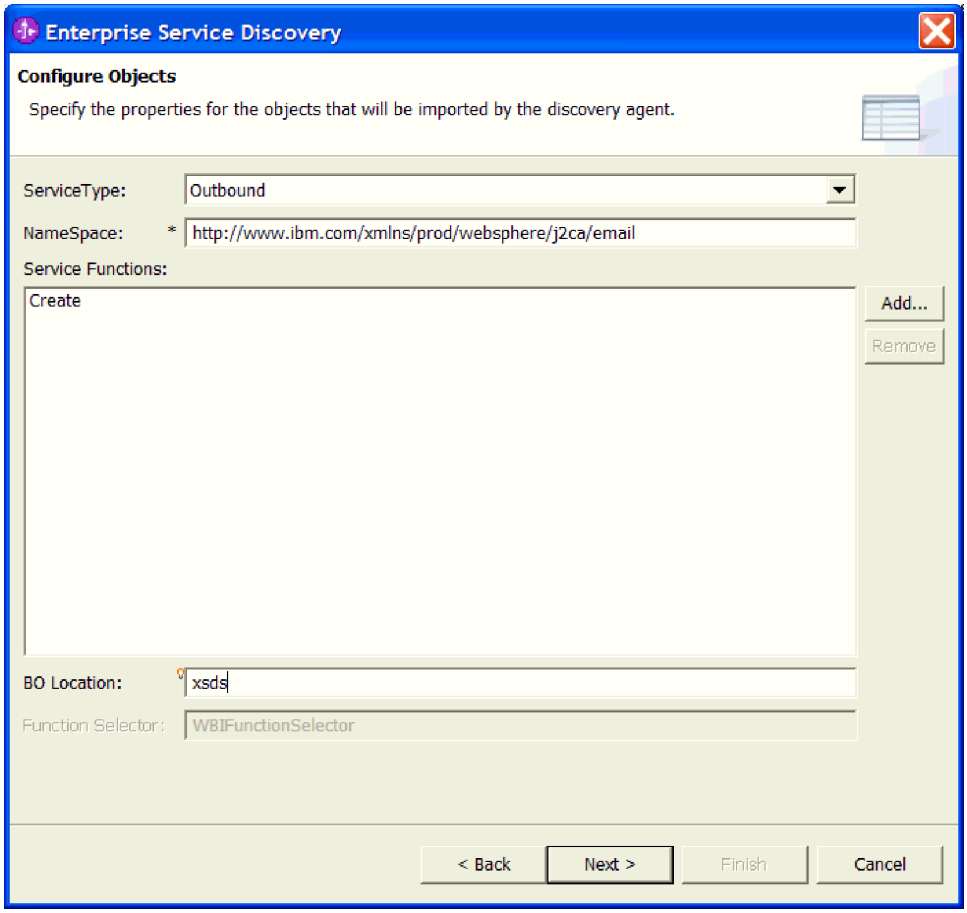
You must have added business objects to the adapter project.

About this task

To configure the selected business objects, follow this procedure.

How to perform this task

1. Select **Outbound** from the **Service Type** drop down menu.
2. Leave the **NameSpace** value unchanged.
3. Type xsds in the **BO Location** field and click **Next**. This is the sample folder where the xsd files with annotations are stored.



Configure Objects window

Result

The objects are configured for outbound communication.

What to do next

Generate outbound artifacts for the adapter project.

Generating artifacts

In this step, the module will be created to store the outbound artifacts.

Before you begin

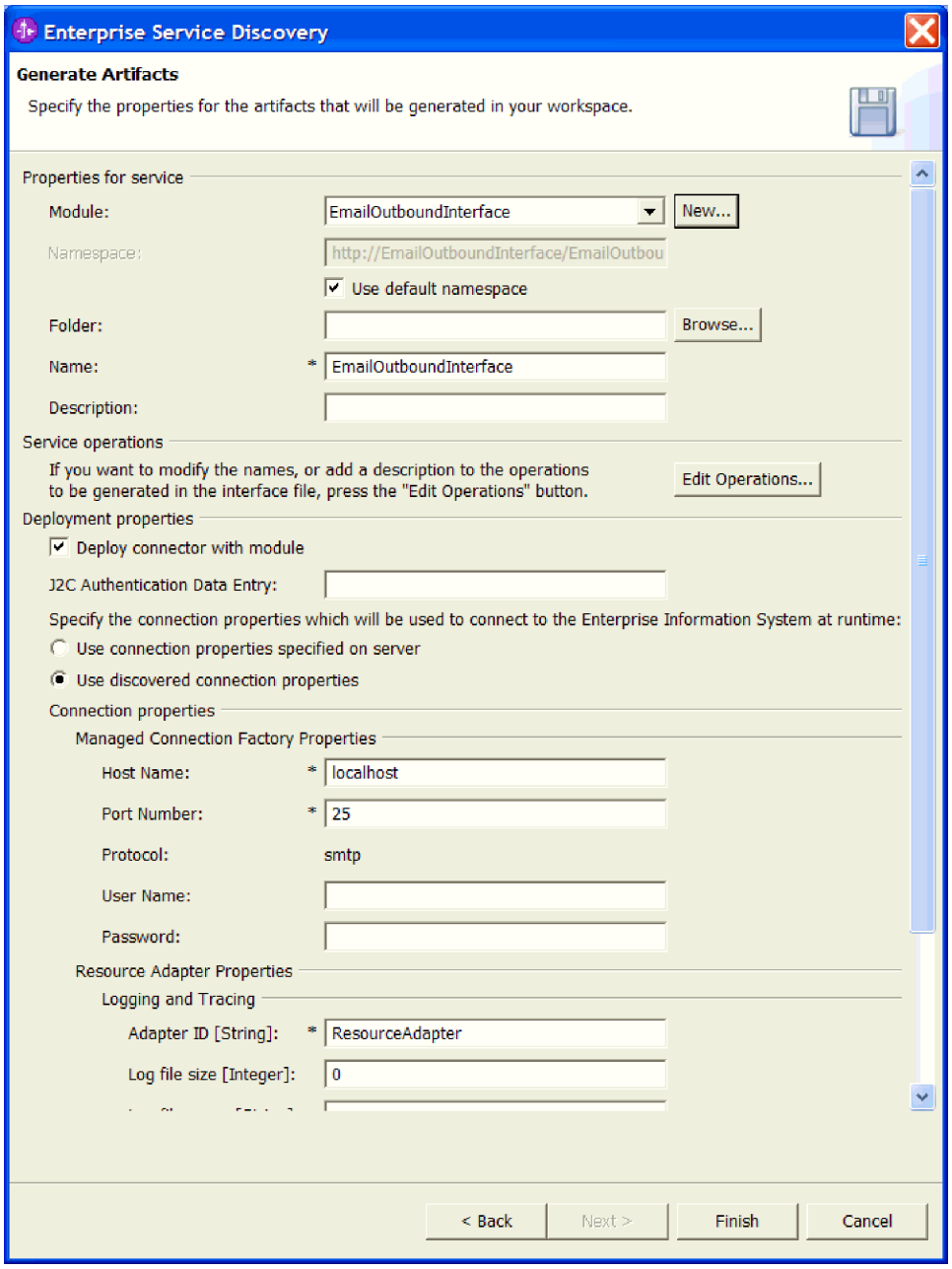
To generate business objects and artifacts for your connector project, you must have already selected business objects to add to the adapter project.

About this task

To generate artifacts for the adapter project, follow this procedure.

How to perform this task

1. Next to the **Module** field, select **New**. The New Integration Project window opens.
2. Select **Create a module project** and click **Next**.
3. Name the module EmailOutbound and click **Finish**. The module is created and a new screen opens to define the managed connection factory properties.
4. **Optional:** To modify the Service operations names or description, click **Edit Operations**.
5. Leave the **Deploy connector with module** check box selected and click **Use discovered connection properties**.
6. **Optional:** Change the default **Hostname** and **Port Number** values to reflect your mail server setup.
7. Change the **Username** and **Password** values to reflect your mail server setup. The UserName and Password must be valid for the e-mail server running on the specified host.
8. **Optional:** Change the **Log file name** and **Trace file name** paths to store the log and trace files in a different location.



Generate Artifacts window

9. Click **Finish** when all of the values have been entered.

Result

The enterprise service discovery wizard will generate the artifacts. The outbound artifacts that are created are visible in WebSphere Integration Developer Project Explorer under EnterpriseApplications/EmailOutbound module.

What to do next

Generate reference bindings for the adapter module.

Generating reference bindings

Create a reference in the assembly editor from the adapter project to a stand-alone reference. A stand-alone reference represents an a generic J2EE component, such as an application server. By wiring the adapter project to a stand-alone reference, you link the adapter to other server processes.

Before you begin

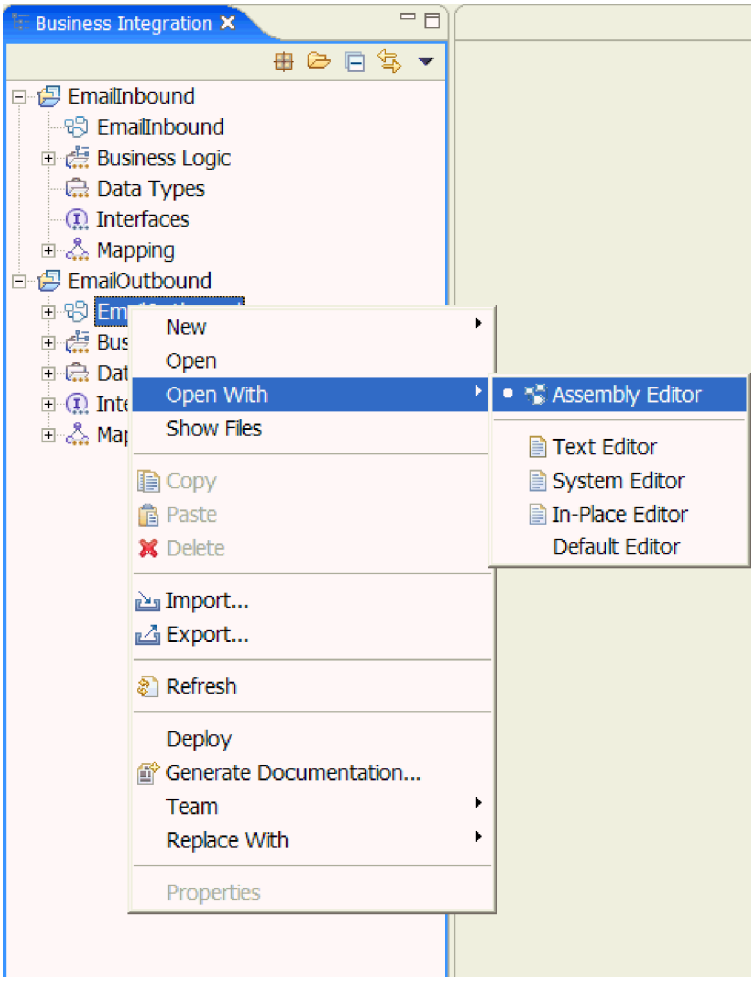
You must have generated artifacts for your adapter project before creating reference bindings.

About this task

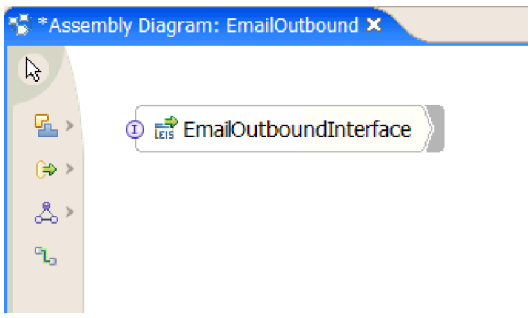
To generate reference bindings, follow this procedure.

How to perform this task

1. **Optional:** Switch to the business integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration Perspective** .
2. Right-click the outbound project and select **Open With** → **Assembly Editor** .
   Opening the Assembly Editor

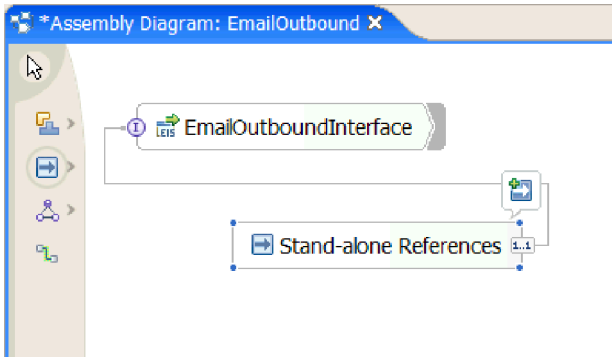


The Assembly editor opens in WebSphere Integration Developer. An icon representing your adapter module interface is present in the assembly editor.



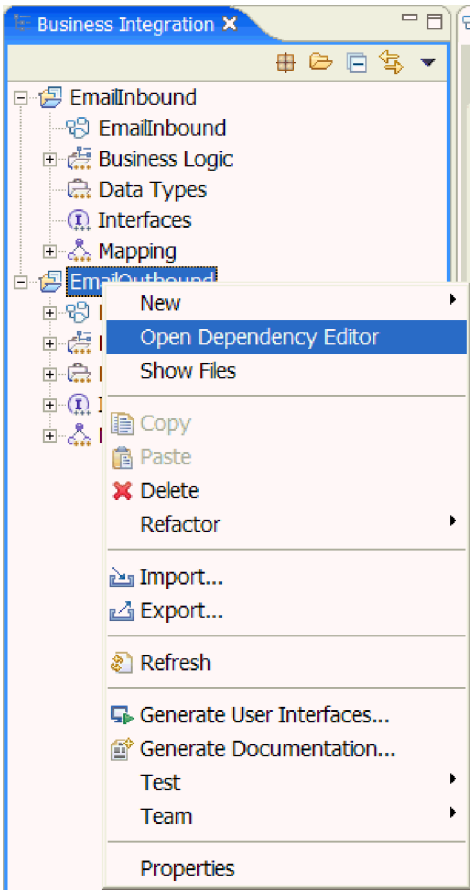
Assembly Diagram window with EmailOutboundInterface representation

3. In the assembly editor, identify and click the **Import** icon. It is a green arrow.
4. Identify and click the **Stand-alone References** icon (blue arrow) and drag it into the assembly editor.
5. Create a wiring from the stand-alone reference to the adapter project by clicking the side of the stand-alone reference representation and dragging the wire to the adapter project representation.



Assembly Diagram EmailOutboundInterface wired to Stand-alone References

6. **Optional:** If a message appears telling you that a reference will be added to the source node and asking you to continue, click **OK**.
7. Save the changes.
8. Right-click the business integration module project and select **Open Dependency Editor**.



Opening Dependency Editor

9. Click **Add** to add your J2EE project.
10. Select your project from the list and save.

Result

A reference binding is created linking the adapter to application server processes.

What to do next

Export the adapter project to an EAR file. The EAR file can be deployed to the application server.

Exporting the module as an EAR file

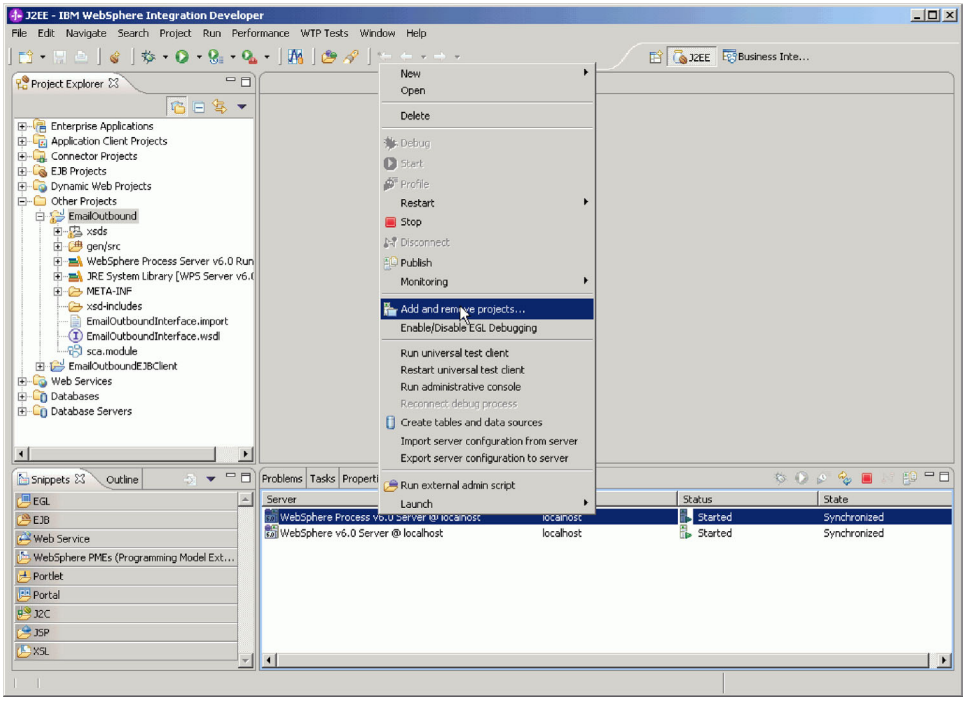
After you have created an outbound module in the enterprise service discovery wizard, deploy it to the test server. This task is optional and is only needed in case you must use the following troubleshooting procedure.

Before you begin

You must have created an outbound module in the enterprise service discovery wizard and created a reference binding.

How to perform this task

1. **Optional:** Confirm that the server is started.
2. Right-click the server instance and select **Add and remove projects**.



Adding the EmailOutbound module to the project

3. Select EmailOutbound and click **Add**. The EmailOutbound module is listed in the Configured projects field.
4. Click **Finish**.

Result

The application is deployed to the test server and started.

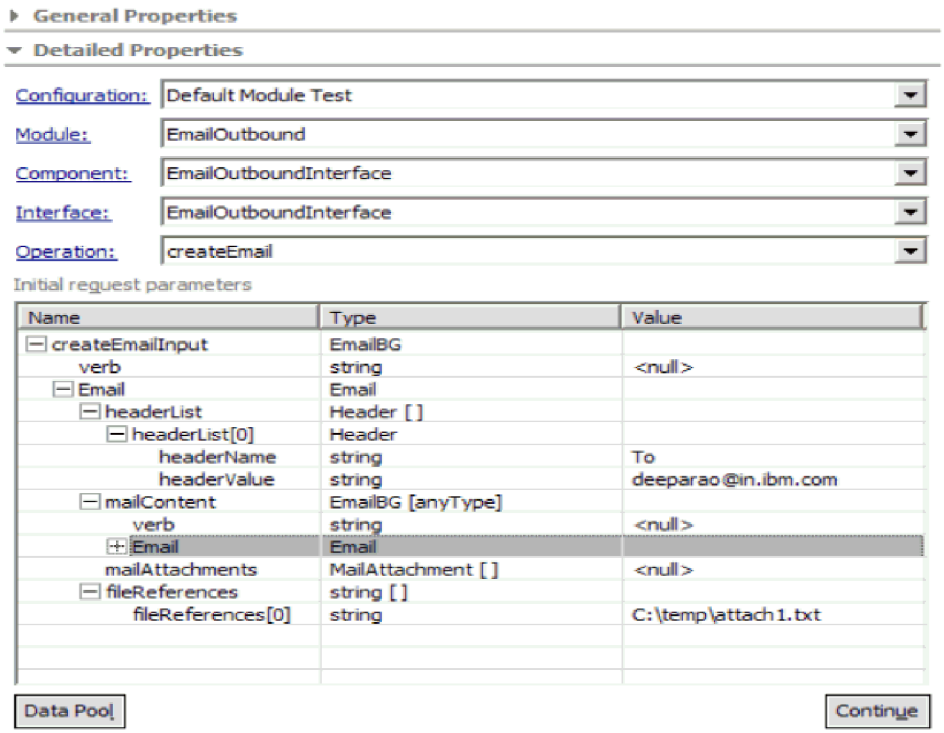
What to do next

Once the application is started, verify the outbound functionality of the adapter using the WebSphere Integrated Test Environment.

Running the module in the test environment

Confirm that your adapter project works by running the EmailOutbound module in WebSphere Integration Developer's integrated test environment.

1. Switch to the business integration perspective.
2. Right-click EmailOutbound in the Project Explorer and select **Test → Test component**.
3. If necessary, change the Initial request parameters for your test scenario and click **Continue**. The **...** button is used to open up the data type selection screen where you can select the required business graph or business object.



Changing business object properties

Note:

- For a pass-through scenario, the mailContent(anyType) field must be set to EmailBG and the mailContent field within EmailBG must be set to UnstructuredContent. This also applies for appBo(any type) and the mailAttachment field in EmailBO. Text/binary for pass-through can be set with any pass-through value.
- For a non-pass-through scenario, the mailContent(anyType) field can be set to the respective business graph type. For example, mailContent can be set to WbiWrapperBG and then filled with specific values. This also applies for appBo(any type) and the mailAttachment field in EmailBO.

4. Select WebSphere Process Server v6.0 and click **Finish**.

Result

The integration test client is started. You can verify your results using the Email client or by observing the status in the integration test client.

What to do next

After running the module in the test environment, verify the success of the test.

Verifying the test results

To determine whether your test was successful, use the Email client to verify that your mail was sent by the mail server. You can also check if your test was successful using the JSP client to examine the System.out log or trace.log files under the log/server1 directory.

Verifying test results using the Email adapter client

Check whether your test execution was successful by examining an e-mail sent by the mail server.

If the content is visible as direct content and the attachments (if any) are present, the e-mail delivery was successful.

Result

If your test was successful, you are finished with this tutorial.

What to do next

If your test was unsuccessful, troubleshoot the tutorial.

Verifying test results using WebSphere Integration Developer Test Client

Check whether your test execution was successful using WebSphere Integration Test Client to examine WebSphere Process Server's System.out log file or by examining the traces file.

- Examine the **Events** → **Invoke** area of the test client. If the test was successful, the following content will be visible:

  ```
  com.ibm.j2ca.email.EmailInteraction execute CWYEM0016I: The Email
  resource adapter has successfully sent the e-mail to all recipients.
  ```

- Check the resource adapter traces file at your configured location. The default isC:\ProgramFiles\IBM\ResourceAdapters\Email\adapter\Email\Samples\EmailRAOut.trace.

Result

If your test was successful, you are finished with this tutorial.

What to do next

If your test was unsuccessful, troubleshoot the tutorial.

Custom data bindings

WebSphere Process Server 6.0.2 provides the following sample custom EIS data bindings: Delimited, FixedWidth, and NameValue. The detailed steps for configuring these data bindings are located in *WPS_install_directory*\samples\doc\CustomEISDataBinding\index.html. The source code for the data binding is also provided and can be modified as needed.

Troubleshooting the tutorial

If WebSphere Integration Developer generates any errors while running the sample scenario, deploy the provided EAR file directly onto WebSphere Process Server using the administrative console.

Exporting the project into an EAR file

Before you can redeploy the project directly through the WebSphere Process Server Administrative Console, you must re-create your EAR file. This is highly recommended in order to eliminate the possibility of any build errors as a cause of the deployment problem.

1. Confirm that there are no errors by building the project.

2. In WebSphere Integration Developer, export the project into an EAR file.
   a. Right-click the project and select **Export → EAR file**.
   b. In the EAR Export window, select the EAR project and provide the absolute path (including the EAR file name) for the Destination property.
   c. Select the Export Source files, Overwrite existing file, and Include project build paths and meta-data files check boxes.

Result

The EAR file is saved in the destination folder.

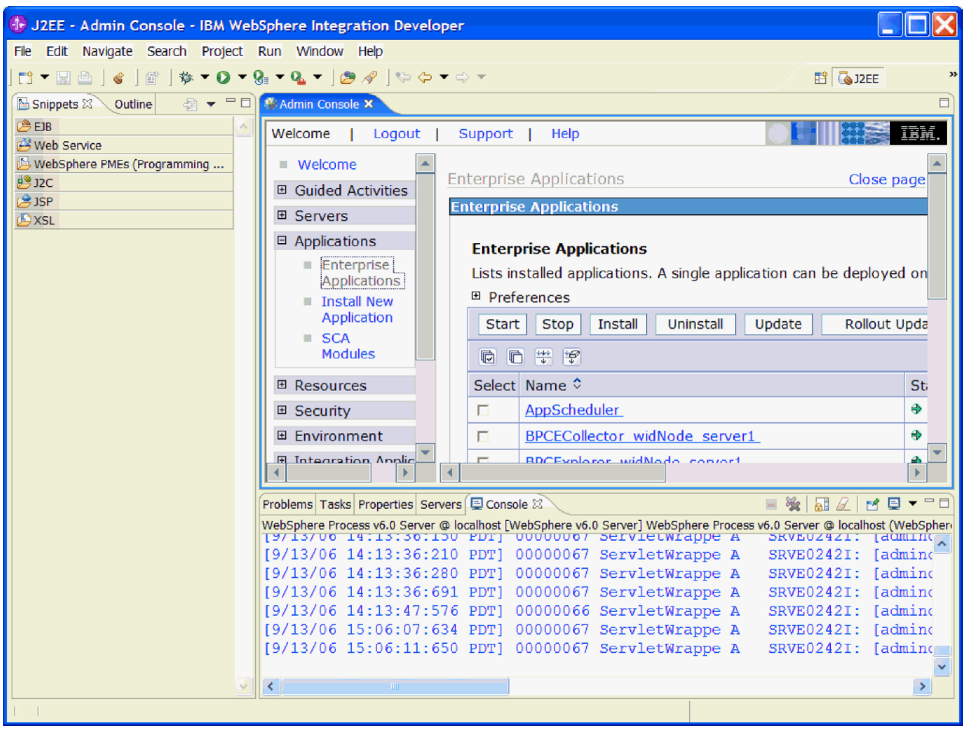
What to do next

If you have successfully exported the project to an EAR file, redeploy the module directly to WebSphere Process Server using the administrative console.

Redeploying the module in the administrative console

After you have recreated your EAR file, redeploy it directly to the test server using the WebSphere Process Server administrative console.

1. Click the Servers tab and verify that the server is started.
2. **Optional:** If it is not already started, right-click on the server instance and start the server.
3. Right-click the server and select **Run adminstrative console**.
4. Either leave the field blank or type Admin in the User ID field and then click **login**. The User ID is used to identify who has logged into the server and is not required.
5. Expand **Applications → Enterprise Applications** and then click **Install**.



Re-installing the EAR file

6. **Browse** for the EAR file and click **Next**.
7. Click **Next** for the Preparing the application for installation window.

8. On the Install New Application window, click **Step 10: Summary** and then **Finish**. A message will confirm when the installation of the application has completed.
9. Click **Save** in the Enterprise Applications window. The application is now installed and saved successfully.

Result

The module is redeployed to the test server.

Preparing to run the inbound scenarios

Before you begin working on either of the inbound sample scenarios, you must create a database to store the events that will be created, configure your data source, confirm that your mail server authentication details are correct for communication with the adapter, and create folders for archiving.

Create the event database

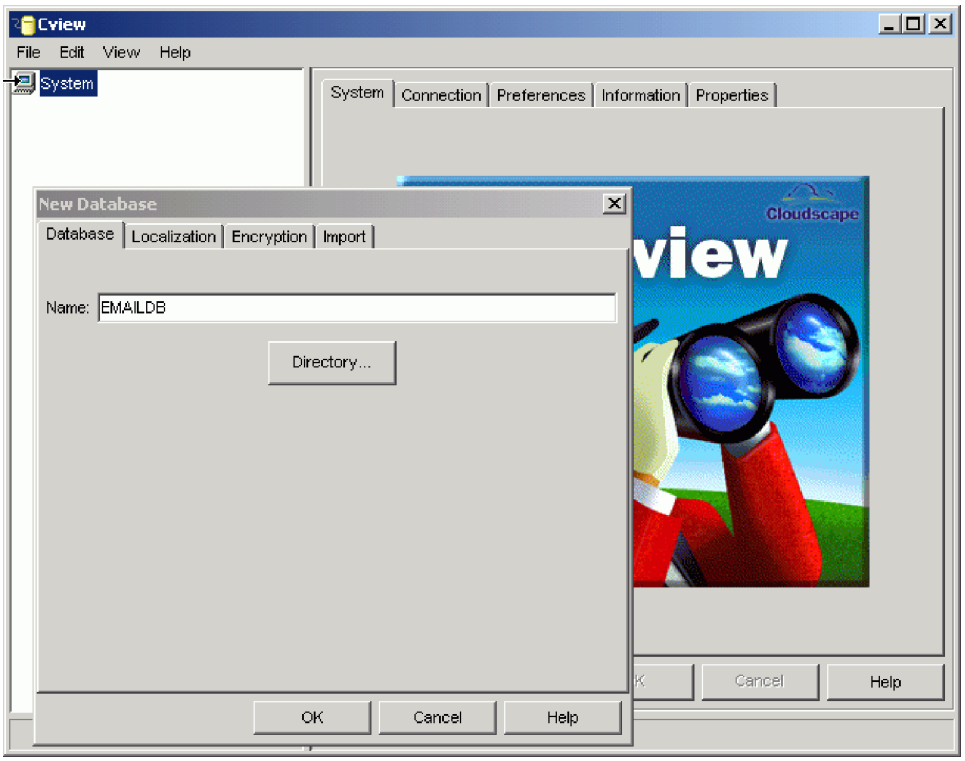
The event database houses the event table, which tracks the progress of events as they move through the system. Once the database is created in Cloudscape™, the event table will be created by the adapter automatically. Install the Cloudscape database supplied by the WebSphere Process Server or Enterprise Service Bus installation.

Before you begin

You must have the adapter installed before creating the event database.

How to perform this task

1. Run the Cloudscape application by browsing to **cloudscape** → **bin** → **embedded** and then double-clicking **cview.bat**.
2. Select **File** → **New** → **Database**.
3. On the **Database** tab, type EMAILDB in capital letters and then click **OK**.



Creating EMAILDB in Cloudscape

4. Close the Cloudscape interface.

What to do next

Configure the data source in WebSphere Integration Developer.

Creating and configuring the data source

Once you have created the EMAILDB database, you must configure WebSphere Integration developer to recognize it as the data source.

1. Open WebSphere Integration Developer. The Business Integration perspective will open by default.
2. Click on the **Server** tab to bring it to the front.
3. Right-click WebSphere Process Server 6.0 and select **Start**. This will start WebSphere Process Server.
4. Click the **Console** tab and confirm that the server is active and ready for transaction requests. When the server is ready the `Server <server profile> is open for e-business` message will be displayed.
5. Open the test server's administrative console window by right-clicking the server profile on the **Server** tab and selecting **Run Administrative Console**. The port number for the administrative console may differ based on your setup. If your port number is different from the default, use your specific port number.
6. In the user ID field, enter the default administrative user ID "admin" and click **Log in**.
7. Create the JDBC data source in WebSphere Process Server with a JNDI name jdbc/EMAILDB.
   a. Open the WebSphere Process Server administrative console by right-clicking the server instance and selecting **Open administrative console.**

b. Select **Resources → JDBC Providers → Cloudscape JDBC Provider → Data sources**.
c. Select new for Data sources.
d. For the **JNDI Name** type jdbc/EMAILDB.
e. For the Database name, type EMAILDB.

8. Return to Data Sources and select the data source you just created. Then, click **Test connection**.

Result

If the connection was successful, `Test connection for data source Cloudscape JDBC Driver DataSource on server 1 at node <node name> was successful` is displayed.

Note: The test connection must be successful to execute the rest of the scenario.

Creating archive folders

For the scenario to properly demonstrate the archiving features of the Adapter for Email, you must set up a series of folders on your system. The In Progress, Archive, and Failed Events folders will be populated by the adapter during various stages of the data transaction.

1. Create a folder called InProgress at C:\EmailRA\Testing\InProgress.
2. Create a folder called Archive at C:\EmailRA\Testing\Archive.
3. Create a folder called FailedEvents at C:\EmailRA\Testing\FailedEvents.

Creating events

The first step in testing the inbound sample application is to create events for the adapter to process. The program supplied with the samples files works for both pass-through and non-pass-through mode, although the events that are created by the sample bat files for pass-through scenario testing will be different from the ones created for non-pass-through scenario tests. The mail format is the same, but the content and attachments will differ. The batch file createEvents.bat provides a command line interface to the EmailEventCreator. If you wish, you may edit the batch file and provide several parameters to create events.

About this task

Artifacts have been shipped on the product CD for creating the tutorial scenario's inbound events. For more information on these artifacts, see the related references.

Parameter values for customizing CreateEvents.bat

Parameter	Value
Host	the IP address of the mail server
Port	Port number
Number of events to be created	Total number of events required for the test
From	The mail ID for the mail account from which the mail was written
To	The mail ID for the mail account to which the mail is to be written

Parameter	Value
CC	The mail ID for the mail account on the cc of the mail
BCC	The mail ID for the mail account on the bcc of the mail
Subject	Subject line content for the mail event
Content	Sample pass-through text content for the mail
Path for event file name	The path to the local folder where the event file attachment (for example, MyAddress.xml) has been stored.
Boolean for passthru/nonpassthru	true or false value indicating whether or not you would like to use a pass-through or non-pass-through scenario. True= pass-through.

To create events using the EventCreator, follow these steps:

How to perform this task

1. Unzip the EmailEventCreator.zip to a <samples_temp> directory.
2. **Optional:** If you want to create custom events, you may edit the following command line parameters in the createEvents.bat file:

   Parameter values for customizing CreateEvents.bat

Parameter	Value
Host	the IP address of the mail server
Port	Port number
Number of events to be created	Total number of events required for the test
From	The mail ID for the mail account from which the mail was written
To	The mail ID for the mail account to which the mail is to be written
CC	The mail ID for the mail account on the carbon copy of the mail
BCC	The mail ID for the mail account on the blind carbon copy of the mail
Subject	Subject line content for the mail event
Content	Sample pass-through text content for the mail
Path for event file name	The path to the local folder where the event file attachment (for example, MyAddress.xml) has been stored.
Boolean for passthru/nonpassthru	true or false value indicating whether or not you would like to use a pass-through or non-pass-through scenario. True= pass-through.

   - For pass-through, enter: `java EmailEventCreator localhost 125 10 from@in.ibm.com to@in.ibm.com cc@in.ibm.com bcc@in.ibm.com TestEmail TestPassthruContent C:\EmailRA\Samples\MyPicture.jpg true` will create 10

events from the user from@in.ibm.com on the localhost mail server with the content from the C:\EmailRA\Samples\MyPicture.jpg attached to the mail. The other attributes of the mail specified by the user will be incorporated on each mail event created by the bat file.

- For non-pass-through, enter: `java EmailEventCreator localhost 125 10 from@in.ibm.com to@in.ibm.com cc@in.ibm.com bcc@in.ibm.com TestEmail TestXMLContent C:\EmailRA\Samples\MyAddress.xml false` will create 10 events from the user venkatre@in.ibm.com on the localhost mail server with the content from the C:\EmailRA\Samples\MyAddress.xml attached to the mail. The other attributes of the mail specified by the user will be incorporated on each mail event created by the bat file.

3. Run either the createEvents.bat file you edited or one of the sample createEvents.bat files supplied to you by double-clicking the file.
4. Verify that the events were transmitted successfully using the mail client before starting the adapter.

Tutorial 3: Receiving inbound data from the EIS using the supplied EAR file

In this tutorial you will deploy a sample EAR file to the test server. Since the EAR file is supplied, you are not required to run the enterprise service discovery wizard to generate your artifacts. This exercise demonstrates how the system can be set up to trigger an automated flow of operation when a customer or automated sends an e-mail containing one or more predefined keywords in the subject field.

About this task

For example, a customer sending an e-mail with the subject "Address change" triggers a series of events in the adapter resulting in an address change. The adapter features demonstrated in this scenario include: inbound event processing capability of the adapter, data transformation capability for known mime-types, data transformation of attachments, search feature for selective polling based on the Subject values, use of the IMAP protocol, and archival of events.

Running the sample application

Once you have created your event database, added your data source, configured your mail server and mail client, and created some events to use with the adapter, you are ready to deploy the sample EAR file to the test server.

1. Switch to the Business Integration Perspective.
2. **Optional:** If the server isn't running, right-click the Servers tab and select **Start**.
3. Select **File → Import**.
4. Browse for **EmailInboundSampleApp.ear** and then click **Finish**. If a dialog asks you if you would like to switch to the J2EE perspective, click **OK**.
5. Right-click the server instance and select **Add and remove projects...**.
6. Select **EmailInboundSampleApp** and click **Add**. The application will be added and visible in the panel on the right.
7. Click **Finish**. The traces noted below on the SystemOut log file confirm the deployment of the application.

Result

The application will be deployed on WebSphere Process Server and start polling for mail server automatically.

Note: If there are any problems installing the application using the above approach, you can install and start the application through the WebSphere Process Server administrative console.

What to do next

Verify the test results using WebSphere Integration Test Client or by checking the failed events folder.

Configuring activation specification properties for mail server connectivity

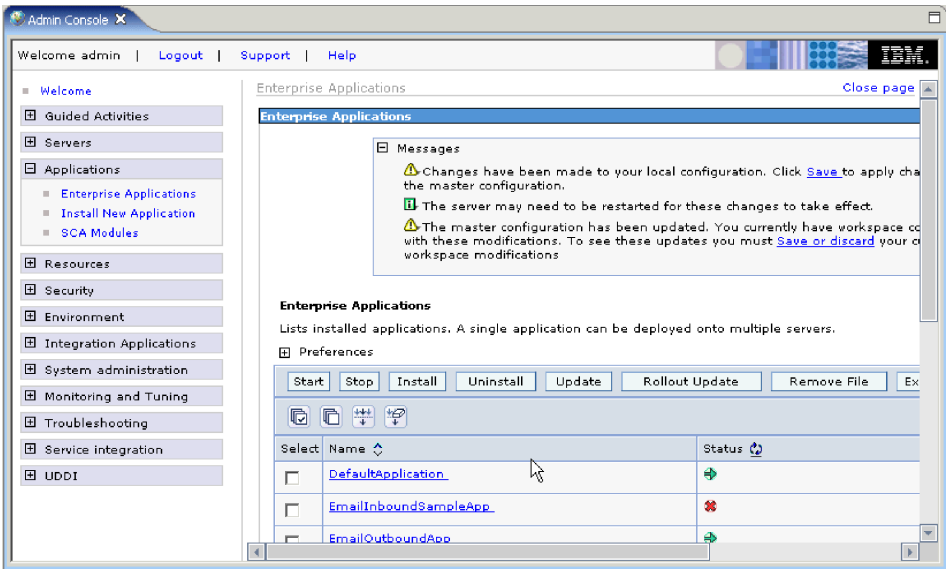
For the adapter to access the mail server, you must change the username and password values in the activation specification properties through the WebSphere Process Server administrative console. In addition, if you are using a remote mail server, you must also change the host and port values.

About this task

If you have not installed the mail server on the same system as WebSphere Process Server or Enterprise Service Bus, set the host and port number properties in the managed connection factory custom properties from the WebSphere Process Server Administrative console as noted in the following steps. Regardless of where your mail server is installed, you must configure two users on your mail server for the adapter project. To do this, follow the steps listed below.

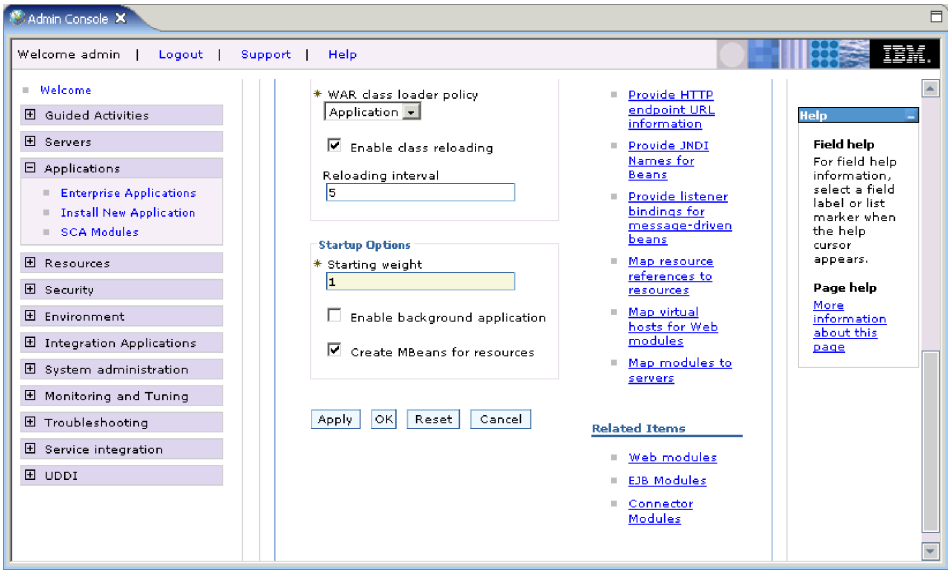
How to perform this task

1. In WebSphere Integration Developer, switch to the J2EE perspective by selecting **Window → Open perspective → Other → J2EE Perspective.**
2. Confirm that the server is started. If it is not, right-click the server instance and start the server.
3. Open the server administrative console by right clicking the server instance and selecting **Run administrative console**.
4. In the admin console, expand **Applications** and click **Enterprise Applications**.



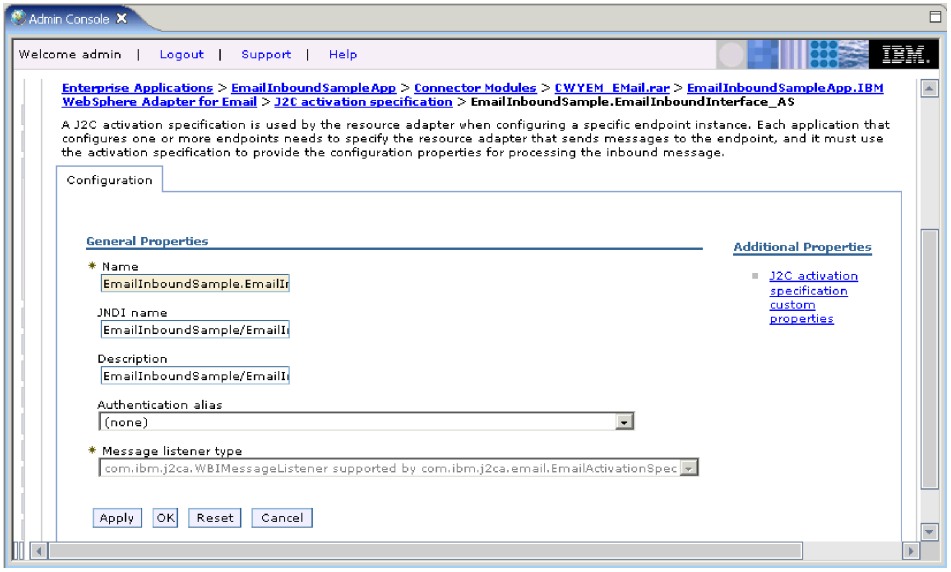
Enterprise Applications window

5. Click **EmailInboundSampleApp** and then scroll to the very bottom of the screen and select **Connector Modules**.

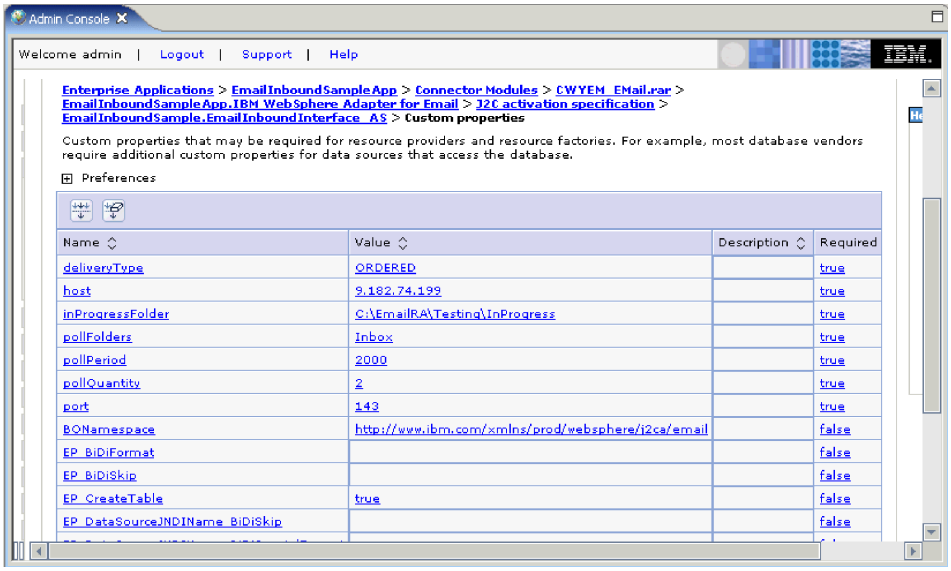


Select Connector Modules

6. Select **CWYEM_Email.rar** and then, under Additional Properties, select **Resource Adapter**.
7. Under Additional Properties, select **J2C Activation specifications**.
8. Select **EmailInboundSample.EmailInboundInterface_AS**.
9. Under Additional Properties, select **J2C activation specification custom properties**.



Select J2C activation specification custom properties The first page of Custom Properties appears. From these screens, you can select and then change any of the managed connection factories property values.



Select and change properties in the admin console

10. Click **userName** to open the userName property dialog and change the mail server name. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
11. **Optional:** Click **password** to open the password property dialog and change the mail server password. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
12. **Optional:** If you are using a remote mail server, click **host** to open the host property dialog and change the mail server host. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.
13. **Optional:** If you are using a remote mail server, you may need to change your port. If so, click **port** to open the port property dialog and change the mail server port. When finished, click **Apply** and then **OK**. You will be returned to the Custom Properties screen.

14. Save your configuration changes.

Result

The adapter has all of the information it will need to connect with the mail server.

Verifying the test results

To determine whether your test was successful, use the Email client to verify that your mail was sent by the mail server. You can also check if your test was successful using WebSphere Integration Test Client to examine the System.out log file or by checking if any failed e-mails have been transferred to the failed events archive folder.

Before you begin

You must have exported the inbound adapter project to an EAR file and deployed it to the application server.

About this task

To verify that your test was successful, look in the trace file or the SystemOut.log file for evidence that your event has been delivered. Or, you can check the failed events archive folder for any e-mails that failed to process.

- If the following message is displayed at <*installation directory*>\Samples\EmailInboundSampleTrace.trc, the e-mail event was successful. `0000003e ResourceAdapter A com.ibm.j2ca.extension.eventmanagement.internal.EventSender deliverEvent() CWYBS0505I: The event has been delivered.`
- If the following message is displayed in WebSphere Process Server's SystemOut.log file located at <*WebSphere Process Server installation directory*>\profiles\<profile_name>\logs\<server_name>, then your e-mail event was successful. `The event has been delivered.`

Result

If your test was successful, you are finished with this tutorial.

What to do next

If your test was unsuccessful, troubleshoot the tutorial.

Troubleshooting the tutorial

If you had problems completing the scenario successfully, consider the troubleshooting tips in this section.

Installing the sample EAR file

Before you can redeploy the project directly through WebSphere Process Server's administrative console, recreate your EAR file. This is highly recommended in order to eliminate the possibility of any build errors as a cause of the deployment problem.

1. In the WebSphere Process Server administration console Welcome admin tab, browse to **Applications → Enterprise Applications**.
2. In the Enterprise Applications panel, select the option **Install**.
3. **Browse** for EmailInboundSampleApp.ear and click **Next**.

4. Click **Next** until you reach the installation summary. Then click **Finish**. During the installation process that follows, you can see the different components of the EAR file getting deployed. When the entire installation gets completed, you will see the message "Application <appname>installed successfully."
5. Click **Save to Master Configuration**.

Result

The EAR file is reinstalled.

Other troubleshooting tips

If reinstalling the EAR file directly to WebSphere Process Server administrative console does not fix your deployment problems, consider these other troubleshooting tips.

- Cloudscape database related – Please note that the Cloudscape embedded database application is single threaded. You will not be able to modify or delete events when WebSphere Process Server is running. WebSphere Process Server must be stopped when you are accessing Cloudscape.
- Known issue: If you deploy and start the adapter without restarting , the adapter might not poll correctly.

Tutorial 4: Creating and exporting an inbound EAR file

In this tutorial you will create your own EAR file and deploy it to the test server. You will see how the adapter processes inbound requests, how data is transformed for known MIME types and attachments, how selective polling works, how events can be archived, and how data is passed through the system. You will also learn about IMAP protocol functions.

Creating the adapter project

To begin the process of creating a module to communicate with an Email service, you create an adapter project. The adapter project contains the adapter itself plus other related artifacts. You create the project by importing the RAR file, which was copied to your local file system during installation, into WebSphere Integration Developer.

About this task

To create an adapter project, use the following procedure.

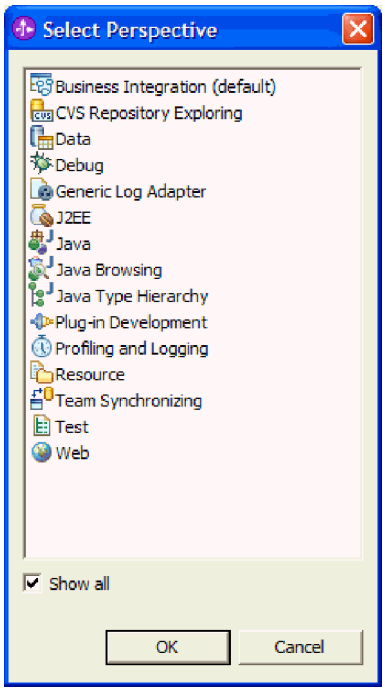
How to perform this task

1. Launch WebSphere Integration Developer by clicking **Start → Programs → IBM WebSphere → Integration Developer 6.0**.
2. If you are prompted to specify a workspace, accept the default value.

   The workspace is a directory where WebSphere Integration Developer stores your project.
3. When the WebSphere Integration Developer window is displayed, close the Welcome page.
4. Switch to the J2EE perspective:
   a. Click **Window → Open Perspective → Other**.
   b. Click **J2EE**.

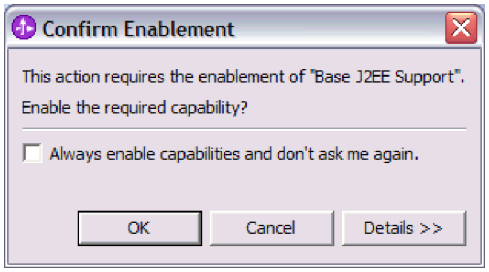
      If **J2EE** is not displayed, select the **Show all** check box, click **J2EE**, and click **OK**.

Figure 11. Selecting J2EE from the Select Perspective list



c. If you see the Confirm Enablement window, select **Always enable capabilities and don't ask me again**.

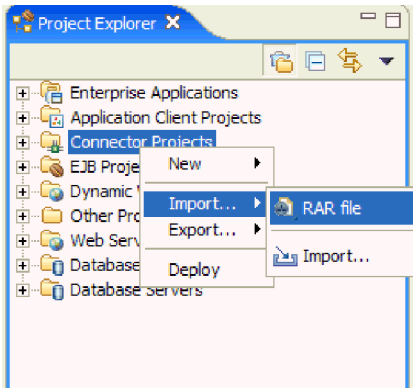
Figure 12. The Confirm Enablement window



d. Click **OK**.

5. Import the RAR file by right-clicking **Connector Projects** and clicking **Import → RAR file**.

Figure 13. Importing the RAR file

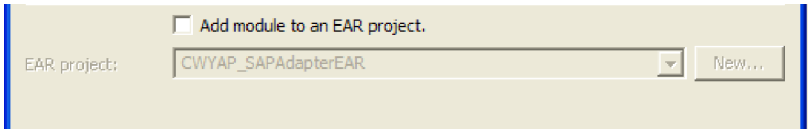


6. Find the RAR file on your local file system by clicking **Browse** and navigating to the directory in which Adapter for Email was installed.
7. Accept the default setting (**CWYEM_Email.rar**) for **Connector project**.

   The connector project has the same name as the RAR file.
8. Accept the default value in the **Target server** field.

   The default value is the test environment for WebSphere Process Server, which is installed as part of WebSphere Integration Developer.
9. Clear the **Add module to an EAR project** check box.

Figure 14. Clearing the Add module to an EAR project check box



   Notice that the EAR project field becomes unavailable after you remove the check mark.
10. Click **Finish**.

Result

The RAR file is imported and a connector project is created. The connector project is visible in the Project Explorer window under Connector Projects.

Configuring the adapter for inbound processing

Use the enterprise service discovery wizard to select business objects and services and to generate artifacts. You can run this tutorial in pass-through or data transformation mode by defining the configuration settings differently.

Setting connection properties for enterprise service discovery

Use enterprise service discovery to determine which services are available to the adapter and to configure the adapter's mail server connection settings.

Before you begin

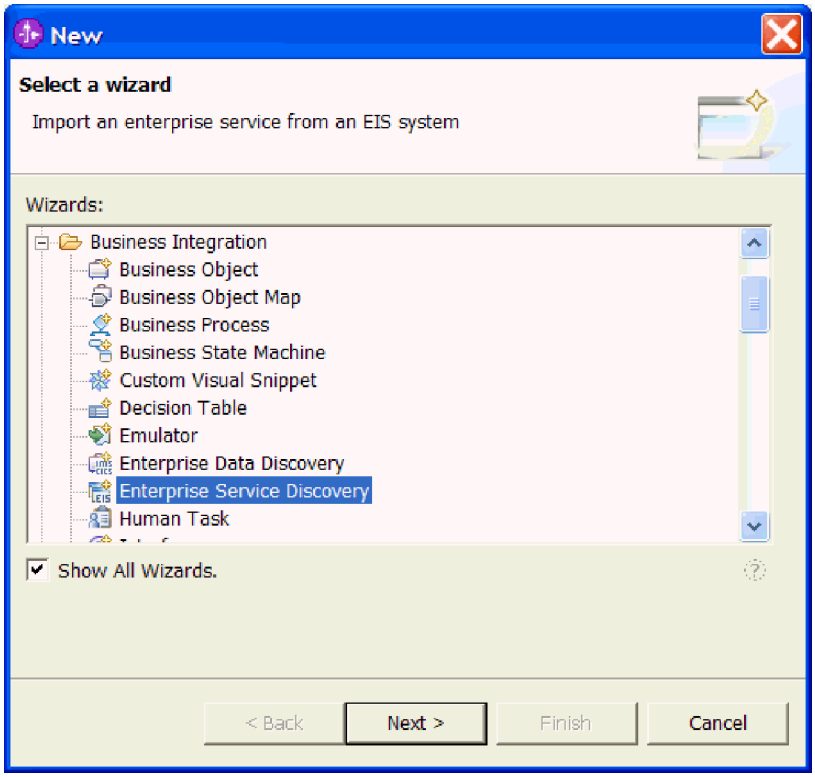
Before you can set the connection properties in this section, you must have created your adapter project.

About this task

To set enterprise service discovery connection properties, follow this procedure.

How to perform this task

1. Change to the Business Integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration**.
2. Select **File** → **New** → **Enterprise Service Discovery**.
3. **Optional:** If Enterprise Service Discovery is not an option, complete the following steps.
   a. Select **File** → **New** → **Other**.
   b. Select the **Show all wizards check box** and then expand **Business Integration**.
   c. Select **Enterprise Service Discovery** and then click **Next**.

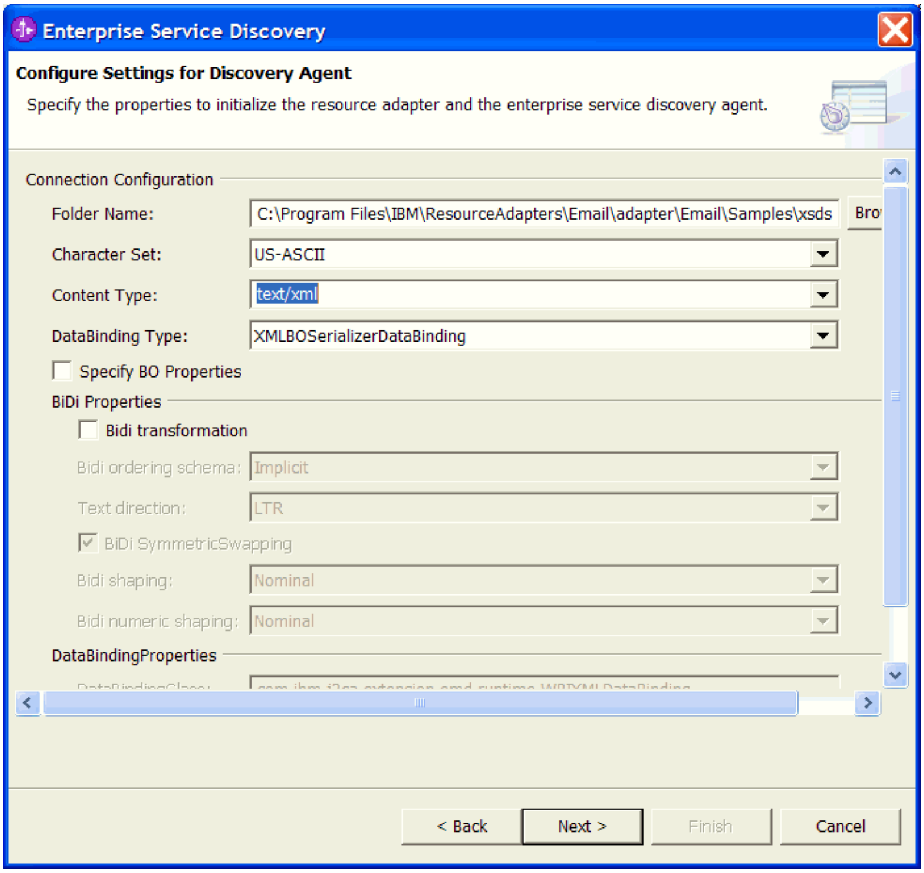


Select a wizard window

4. Select the CWYEM_Email connector project and click **Next**.
5. Make the following selections for the Configure settings for Discovery Agent window:
   a. Browse for the xsd files provided with the sample. The default installation location for the sample xsd file zip is **Program Files** → **IBM** → **ResourceAdapters** → **Email** → **adapter** → **Email** → **Samples**.

      Note: To demonstrate a pass-through scenario (without data transformation), leave this field blank.
   b. Select US-ASCII for the **Character Set**.
   c. Select text/xml for the**Content Type** format. When the content type is selected, the corresponding data binding types fields are populated.
   d. Leave the **BiDi transformation** check box unchecked.

e. **Optional:** To change the log and trace file directory, click the **Show Advanced** button and browse for a new directory.



Configure settings for Discovery Agent window

f. Select **Next**.

Result

Using these properties, the enterprise service discovery wizard will initialize the adapter and the enterprise service discovery agent.

What to do next

Select the business objects and services to associate with the adapter module.

Selecting business objects and services

Run a query to introspect what services are available to the adapter. The enterprise service discovery wizard displays a metadata tree of all the business objects discovered by the query from which you will select which business object to add to the module.

Before you begin

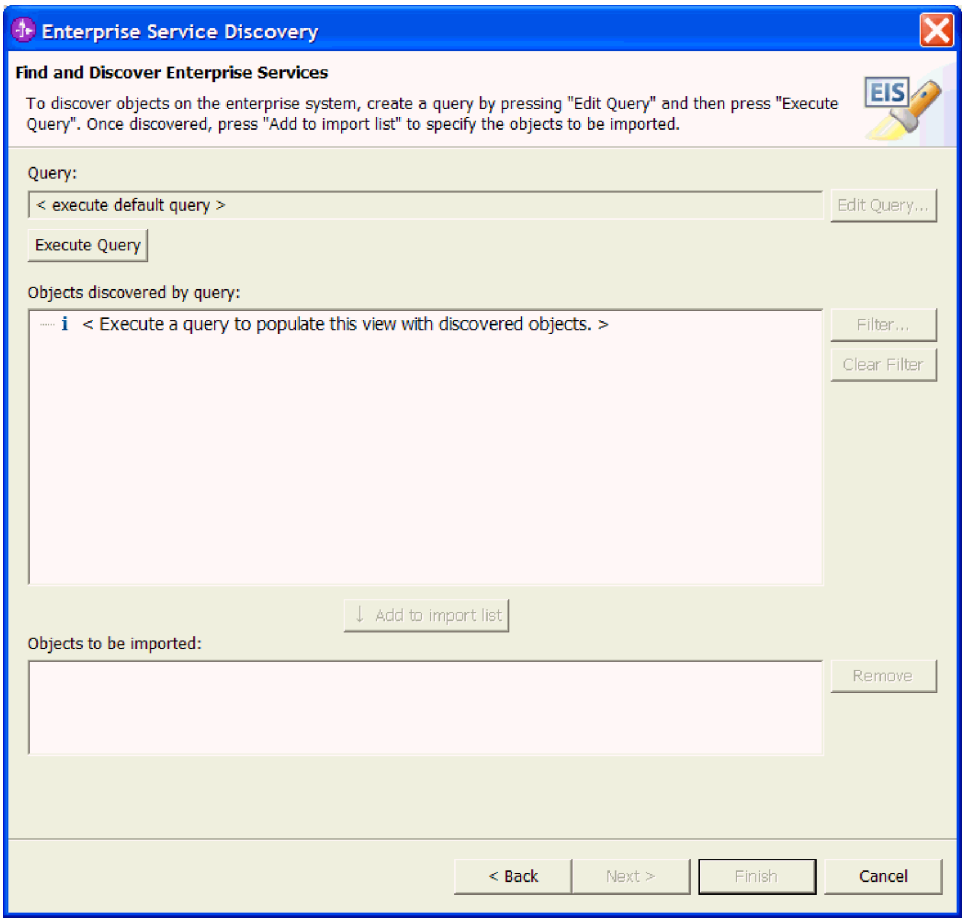
You must have specified the adapter configuration properties for the discovery service before you can complete the steps below.

About this task

To select business objects and services to be used with the adapter, follow this procedure.

How to perform this task

1. Click **Execute Query**. Any business objects found in the folder defined on the last screen are displayed under Objects discovered by query.
2. Select the Wbiaddress business object and click **Add to import list**.



Find and Discover Enterprise Services window

3. The business objects are added to Objects to be imported. Click **Next**.

Result

Using these properties, the discovery service prepares a metadata tree that will be displayed for object selection and navigation in the following steps.

What to do next

Configure the selected business objects.

Configuring the selected objects

Once you have added business objects to the module, configure them for inbound operations.

Before you begin

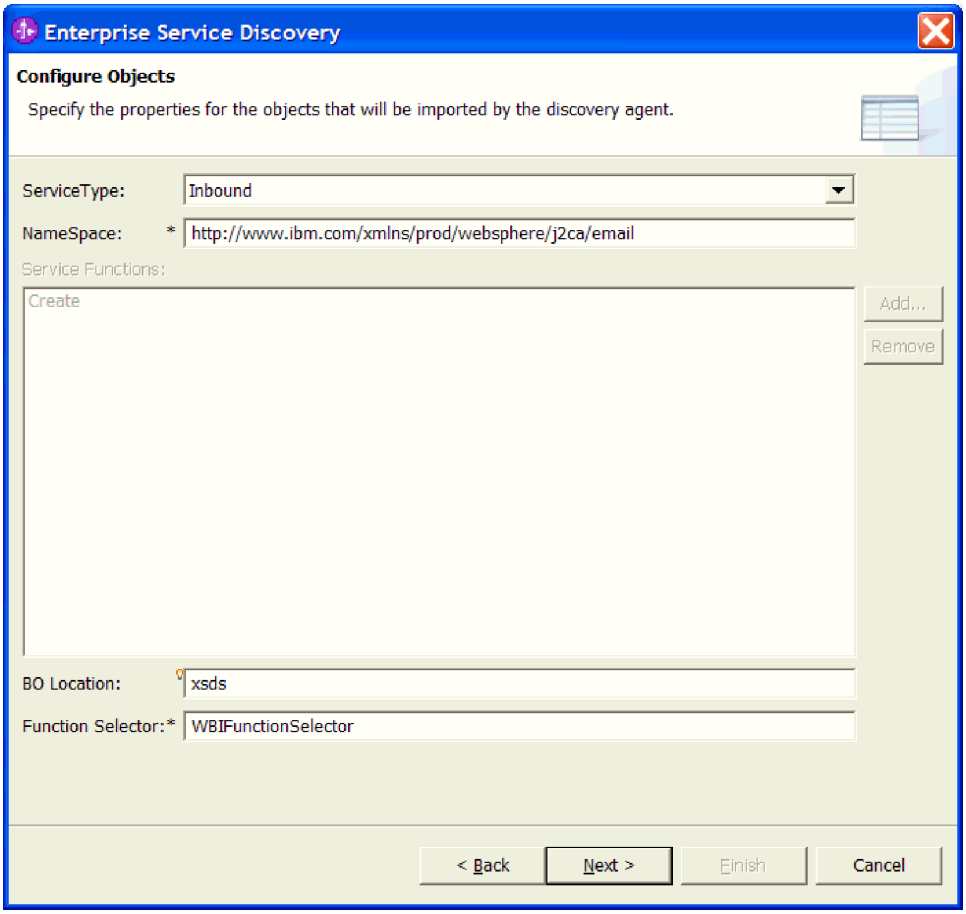
You must have added business objects to the adapter project.

About this task

To configure the selected business objects, follow this procedure.

How to perform this task

1. Select **Inbound** from the **Service Type** drop down menu.
2. Leave the **NameSpace** value unchanged.
3. Type xsds in the **BO Location** field and click **Next**. This is the sample folder where the xsd files with annotations are stored.



Configure objects window

Result

The objects are configured for outbound communication.

What to do next

Generate outbound artifacts for the adapter project.

Generating artifacts

In this step, the module will be created to store the inbound artifacts.

Before you begin

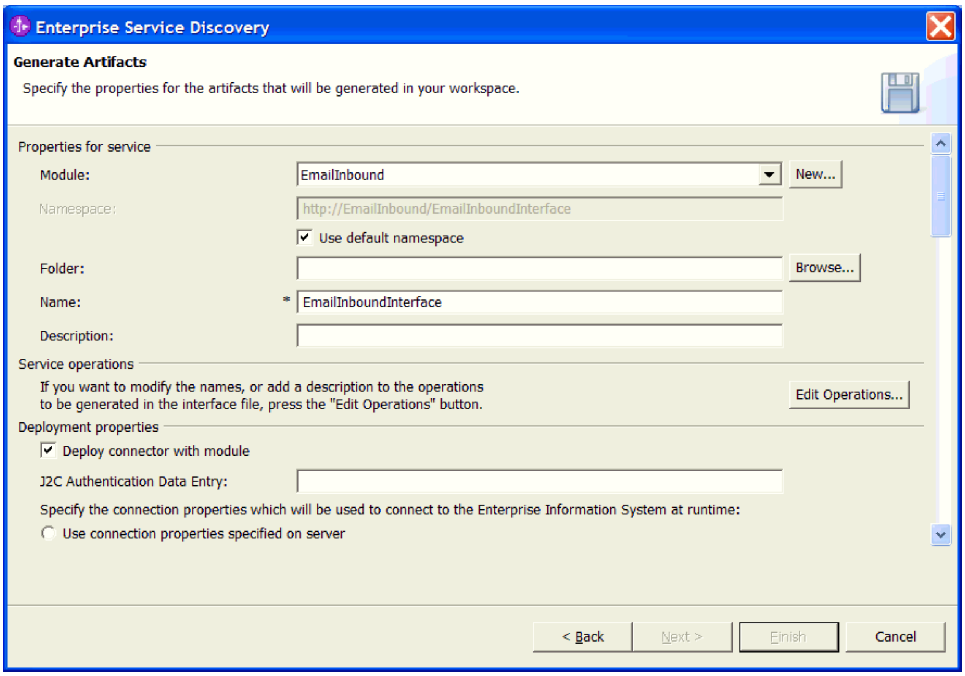
To generate business objects and artifacts for your connector project, you must have already selected business objects to add to the adapter project.

About this task

To generate artifacts for the adapter project, follow this procedure.

How to perform this task

1. Next to the **Module** field, select **New**. The New Integration Project window opens.
2. Select **Create a module project** and click **Next**.
3. Name the module EmailInbound and click **Finish**. The module is created and a new screen opens to define the activation specification properties.
4. **Optional:** To modify the Service operations names or description, click **Edit Operations**.
5. Leave the **Deploy connector with module** check box selected and click **Use discovered connection properties**.
6. **Optional:** Change the default **Hostname** and **Port Number** values to reflect your mail server setup. For more information about event persistence properties, see the activation specification properties in the references section.
7. Change the **Username** and **Password** values to reflect your mail server setup. The UserName and Password must be valid for the e-mail server running on the specified host.
8. **Optional:** Change the **Log file name** and **Trace file name** paths to store the log and trace files in a different location.



Generate Artifacts window

9. Click **Finish** when all of the values have been entered.

Result

The enterprise service discovery wizard will generate the artifacts. The inbound artifacts that are created are visible in the WebSphere Integration Developer Project Explorer under EmailInbound module.

What to do next

Generate reference bindings for the adapter module.

Generating reference bindings

Create a reference in the assembly editor from the adapter project to an endpoint. The endpoint represents an a generic J2EE component, such as the application server. By wiring the adapter project to the stand-alone reference (represented as a component in the assembly editor), you link the adapter to other server processes.

Before you begin

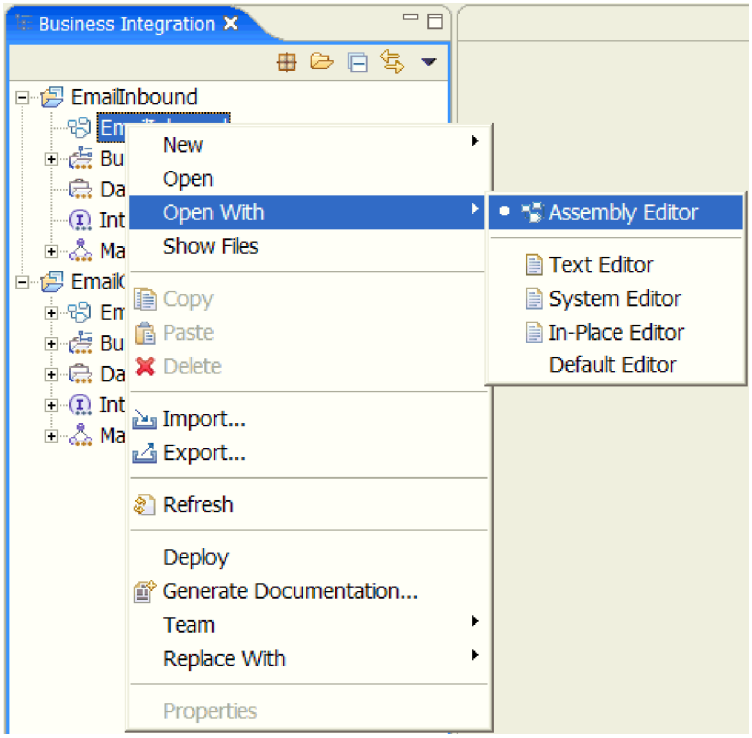
You must have generated artifacts for your adapter project before creating reference bindings.

About this task

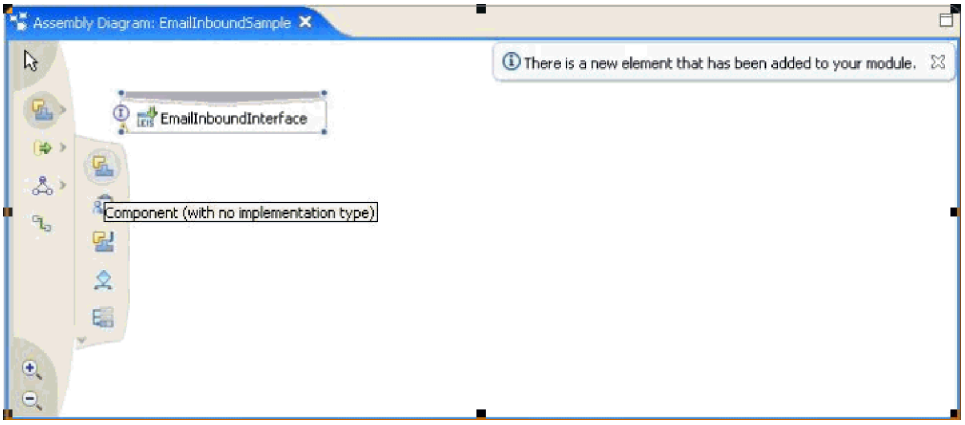
To generate reference bindings, follow this procedure.

How to perform this task

1. **Optional:** Switch to the business integration perspective by selecting **Window** → **Open perspective** → **Other** → **Business Integration Perspective** .
2. Right-click the inbound project and select **Open With** → **Assembly Editor** .
   Opening the Assembly Editor

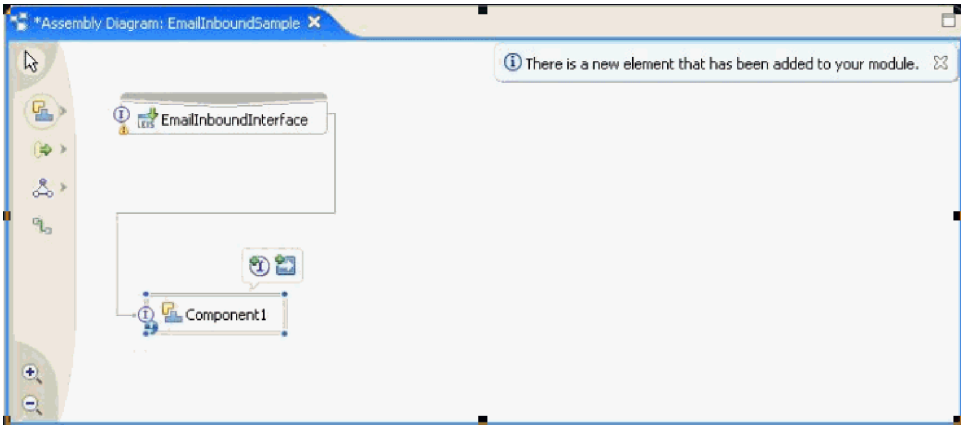


   The Assembly editor opens in WebSphere Integration Developer. An icon representing your adapter module interface is present in the assembly editor.
3. Create a component. The component represents a J2EE client, like an application server, that reads inbound requests from the adapter.
4. Identify and drag the **Component (with no implementation type)** icon into the Assembly Editor.



Assembly Diagram window with EmailInboundInterface representation

5. Click the side of the stand-alone reference representation and drag the wire to the adapter project representation. This draws a wire from the adapter project to the new component.



Assembly Diagram EmailInboundInterface wired to Component 1 representation

6. Click **OK** if any dialog boxes ask you to save the changes.
7. Right-click the component and select **Generate Implementation → Java**.
8. Select the default package and click **OK**. WebSphere Integration Developer generates the Java implementation for the endpoint component and opens it in a window.
9. Save the generated Component1 Java implementation and the assembly diagram.

Result

A reference binding is created linking the adapter to the J2EE client.

What to do next

Configure an inbound adapter module or export the outbound adapter module to an EAR file. The EAR file can be deployed to the application server.

Exporting the module as an EAR file

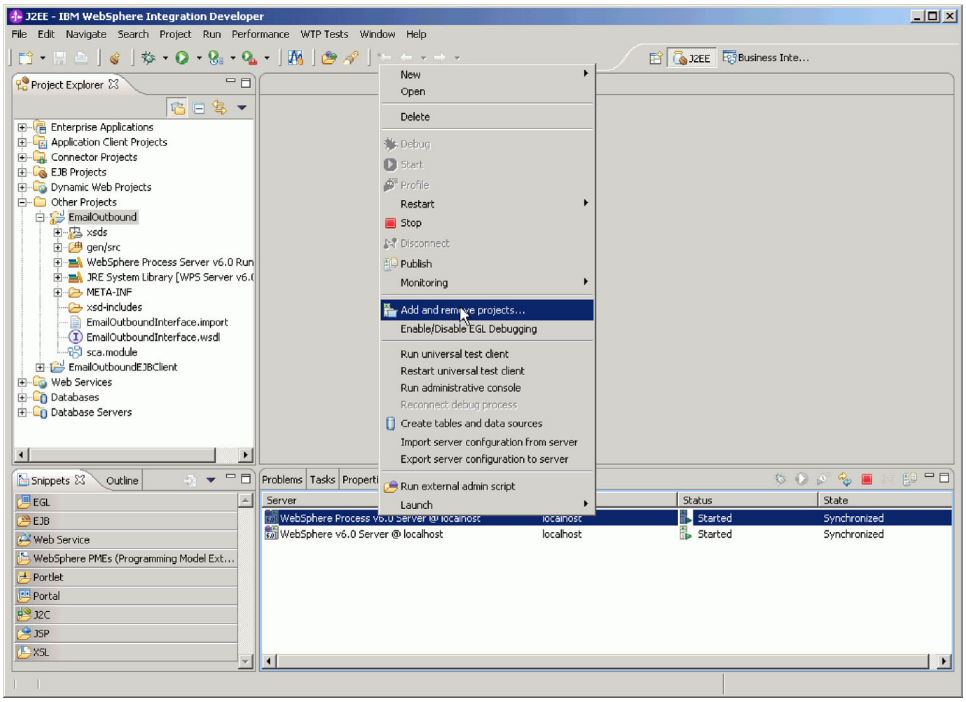
After you have created an inbound module in the enterprise service discovery wizard, deploy it to the test server. This task is optional and is only needed in case you must use the following troubleshooting procedure.

Before you begin

You must have created an inbound module in the enterprise service discovery wizard and created a reference binding.

How to perform this task

1. **Optional:** Confirm that the server is started.
2. Right-click the server instance and select **Add and remove projects**.



Adding the EmailInbound module to the project

3. Select EmailInbound and click **Add**. The EmailInbound module is listed in the Configured projects field.
4. Click **Finish**.

Result

The application is deployed to the test server and started.

What to do next

Once the application is started, verify the inbound functionality of the adapter using the WebSphere Integrated Test Environment.

Verifying the test results

Check if your test was successful using the WebSphere Integration Test Client to examine the System.out log file. Or, you can check if any failed e-mails have been transferred to the failed events archive folder.

Before you begin

You must have exported the inbound adapter project to an EAR file and deployed it to the application server.

About this task

To verify that your test was successful, look in the trace file or the SystemOut.log file for evidence that your event has been delivered. Or, check the failed events archive folder for the presence of any e-mails that failed to process.

- If the following message is displayed at :\EmailRA\Samples\EmailInboundTrace.trc, the e-mail event was successful. `0000003e ResourceAdapter A com.ibm.j2ca.extension.eventmanagement.internal.EventSender deliverEvent() CWYBS0505I: The event has been delivered.`
- If the following message is displayed in WebSphere Process Server's SystemOut.log file located at <*WebSphere Process Server installation directory*>\profiles\<profile_name>\logs\<server_name>, then your e-mail event was successful. `The event has been delivered.`

Result

If your test was successful, you are finished with this tutorial.

What to do next

If your test was unsuccessful, troubleshoot the tutorial.

Custom data bindings

WebSphere Process Server 6.0.2 provides the following sample custom EIS data bindings: Delimited, FixedWidth, and NameValue. The detailed steps for configuring these data bindings are located in *WPS_install_directory*\samples\doc\CustomEISDataBinding\index.html. The source code for the data binding is also provided and can be modified as needed.

Troubleshooting the tutorial

If you had problems completing the scenario successfully, consider the troubleshooting tips in this section.

Installing the sample EAR file

Before you can redeploy the project directly through WebSphere Process Server's administrative console, recreate your EAR file. This is highly recommended in order to eliminate the possibility of any build errors as a cause of the deployment problem.

1. In the WebSphere Process Server administration console Welcome admin tab, browse to **Applications → Enterprise Applications**.
2. In the Enterprise Applications panel, select the option **Install**.
3. **Browse** for EmailInboundSampleApp.ear and click **Next**.
4. Click **Next** until you reach the installation summary. Then click **Finish**. During the installation process that follows, you can see the different components of the EAR file getting deployed. When the entire installation gets completed, you will see the message "Application <appname>installed successfully."
5. Click **Save to Master Configuration**.

Result

The EAR file is reinstalled.

Other troubleshooting tips

If reinstalling the EAR file directly to WebSphere Process Server administrative console does not fix your deployment problems, consider these other troubleshooting tips.

- Cloudscape database related – Please note that the Cloudscape embedded database application is single threaded. You will not be able to modify or delete events when WebSphere Process Server is running. WebSphere Process Server must be stopped when you are accessing Cloudscape.
- Known issue: If you deploy and start the adapter without restarting , the adapter might not poll correctly.

Chapter 12. Viewing the sample adapter artifacts

To view the sample artifacts for each tutorial, import into IBM WebSphere Integration Developer the quick start reference files included with the adapter. Note that these artifacts are for reference only. They probably will not execute in your enterprise information system environment. If you have not stepped through the tutorials, you can still use the reference files to view examples of correctly-generated artifacts before you create your own.

Before you begin

Locate the quick start reference files in the referencefiles subdirectory of the samples directory. There is a project interchange zip file for each quick start tutorial. For instance, Tutorial1.zip is for quick start tutorial 1.

Important: Do not modify or use the artifacts provided in the quick start reference files. They are provided exclusively for viewing.
Reference files do not include third-party libraries. When imported into IBM WebSphere Integration Developer, the reference files might generate compilation error messages because dependent libraries are missing. The artifacts in the reference files may not be compatible with the enterprise information system (EIS) you are using. They vary based on EIS version and configuration.

The artifacts were generated with version 8.95 of JD Edwards EnterpriseOne Tools and version 8.11 of JD Edwards EnterpriseOne.

The artifacts were generated with version 11.5.10.2 of Oracle E-Business Suite.

The artifacts were generated with version ECC 6.0 of SAP Software.

The artifacts were generated with version 7.8.2 of Siebel.

About this task

Import the quick start reference files into WebSphere Integration Developer to view sample artifacts associated with each quick start tutorial.

How to perform this task

1. In the Business Integration perspective of WebSphere Integration Developer, click **File → Import**.
2. In the Import window, select **Project Interchange** and click **Next**.
3. Select the project interchange file containing the tutorial artifacts you want to view.
4. Import all the projects in the project interchange file by clicking **Select All** .
5. Click **Finish**.

Result

A business integration module is created with the following artifacts:

- Service import and export definitions
- Business objects (service data objects)

- Interfaces.

Chapter 13. Reference information

Reference information supports the tasks you want to complete. The information includes all configurable properties for the Email adapter, messages, and related product information.

Enterprise service discovery wizard configuration properties

The properties in this section should be configured using the enterprise service discovery wizard before deployment or with the WebSphere Application Server administrative console after deployment.

Enterprise service discovery connection properties

Enterprise service discovery connection properties include outbound and inbound connection properties required for performing metadata discovery and bidirectional configuration. You can configure these properties using the enterprise service discovery wizard when you initially deploy the adapter or through the WebSphere Process Server administration console after deployment.

When you run the enterprise service discovery wizard in WebSphere Integration Developer, specify the connection properties listed below.

Table 5. Enterprise service discovery connection properties

Property	Description	Required	Default value	Property type
ContentType	Content type (for example, text/xml or text/namevalue) to be used for all imported business objects.	No	Empty field	Editable list of values (formats supported by the existing data handlers)
DataBindingType	Name of the data binding corresponding to the content type.	No	None	Editable list of values (WBIA DataHandler DataBinding, XML DataBinding)
DataBindingProperties	Property group for the selected data binding type.	No	Null	Property group (The list of properties is dependent on the selection of specific data binding information in the Data Binding Properties field)
FolderName	Folder on your local system where the xsd files are located.	Yes	None	WBIFolderProperty

Table 5. Enterprise service discovery connection properties (continued)

Property	Description	Required	Default value	Property type
SpecifyIndividualBOprops	Select this in the wizard to specify properties for individual object selection.	No	False	Boolean

Enterprise service discovery selection properties

Enterprise service discovery selection properties are used to build a service description and to save the built-in artifacts. These properties are configured in the enterprise service discovery wizard.

When you run the enterprise service discovery wizard in WebSphere Integration Developer, specify the enterprise metadata discovery selection properties listed below.

Table 6. Enterprise service discovery enterprise metadata discovery selection properties

Property	Description	Required	Default	Property type	Globalized
Service Type	This property represents the service type. This property is read-only. Values for this property are Inbound or Outbound	Yes	Outbound	String	No
NameSpace	This represents the value of the default namespace	Yes	http://www.ibm.com/smlna/prod/websphere/j2ca/email	String	No
Bo Location	This is the relative path to the location where the generated business objects will be saved.	No	None	String	Yes
Operations	The adapter only supports the create operation for outbound communication. Inbound does not have any supported operations.	No	CREATE (outbound)	String	No
Function Selector	Name of the Function Selector implementation (available only when the ServiceType= inbound)	Yes	WBIFunction Selector	String	No

Enterprise service discovery object selection properties

Enterprise service discovery object selection properties allow you to specify in the enterprise service discovery wizard whether they would like to use the default business object properties or define the values for each individual business object in the adapter project.

When you run the enterprise service discovery wizard in WebSphere Integration Developer, specify the object selection properties listed below.

Table 7. Enterprise service discovery object selection properties

Property	Descriptions	Required	Default	Property type	Globalized
ContentType	Editable list of values (formats supported by datahandlers)	No	None	String	No
DataBindingType	Editable list of values (WBIDataHandler DataBinding, XMLDataBinding)	No	None	String	No
Data Binding Properties	The list of properties is dependent on the selection of specific data binding in the DataBindingType field	No	None	Property group	No
Expose For Service Description	Determines if the selected object has to be exposed in the service description as an input/output type. This property will always be set to a false value for the adapter since content-specific wrappers are not supported.	No	false	Boolean	No

Adapter configuration properties

The properties in this section should be configured using the enterprise service discovery wizard before deployment or with the WebSphere Application Server administrative console after deployment.

Resource adapter properties

Resource adapter properties consist of logging and tracing, bidirectional language support, and activities specific to the adapter, such as the default configuration properties of the adapter. You configure these properties using WebSphere Process Server administrative console.

When you configure the adapter, specify the resource adapter properties listed below.

Table 8. Resource adapter properties

Property	Description	Required	Global-ized	Bidi support	Default
Resource adapter class	The resource adapter implementation class	Yes	No	No	com.ibm.j2ca.email.EmailResource Adapter
managed connection factory class	The managed connection factory (MCF) interface for the adapter	Yes	No	No	com.ibm.j2ca.email.EmailManaged ConnectionFactory
connection factory interface	The connection factory implementation class	Yes	No	No	javax.resource.cci.ConnectionFactory
connection factory impl class	The connection factory implementation class	Yes	No	No	com.ibm.j2ca.email.Email ConnectionFactory
connection interface	The connection interface name	Yes	No	No	javax.resource.cci.Connection
connection impl class	The connection implementation class	Yes	No	No	com.ibm.j2ca.email.EmailConnection
enableHA Support	When the enableHA Support property is set to true, only one of the replicated adapter instances actively polls for events while other instances are in standby mode. If the enableHA Support property is set to false, all of the adapter instances replicated on cluster members actively poll for events. This may result in event duplication. Do not change the value of enableHA Support to false for single server environments.	No			True

Managed (J2C) connection factory properties

Managed connection factory configuration properties are used at run time to create an outbound connection instance with an enterprise information system.

When you configure the adapter, specify the properties listed below.

Note: The ESD wizard refers to these properties as managed connection properties and the WebSphere Process Server refers to these as (J2C) connection factory properties.

Table 9. Managed (J2C) connection factory properties

Property	Required	Description	Type	Globalized	Bidi support	Default
Host	Yes	The host IP address for the mail server	String	No	No	localhost
Port	Yes	The port on which the mail server is listening	String	No	No	25
Protocol	Yes	The protocol to be used for outbound communication with the mail server. For this release, SMTP is the only protocol supported for outbound communication.	String	No	No	SMTP
Username	No	The user ID to be used during the SMTP session	String	No	Yes	None
Password	No	The password associated with the user ID used to authenticate the SMTP session.	String	No	Yes	None

Data transformation properties

Data transformation properties enable your adapter to convert unstructured data formats to WebSphere Business Integration business objects and to convert business objects back to unstructured data formats. They can be set through the enterprise service discovery wizard.

When you configure the adapter, specify the data transformation properties listed below.

Table 10. Data transformation properties

Property	Required	Description	Bidi support	Globalized	Default value
Function Selector	Yes	Name of the Function Selector implementation class. **Note:** This is only available if the ServiceType= inbound	No	No	WBIFunctionSelector

Activation specification properties

Activation specification properties hold the inbound event processing configuration information for a message endpoint. They can be set through the enterprise service discovery wizard or the WebSphere Process Server administrative console.

When you configure the adapter, specify the activation specification properties listed below.

Table 11. Activation specification properties

Property	Required	Description	Default	Example
ArchiveFileName	No	The user can specify the pattern used to name the archive file by creating a comma delimited list of header names.	The mail message-ID retrieved from the mail header.	Example: *From, Date* becomes From + Date + MessageID. In the example, the adapter takes the From address and the Date from the mail headers and then forms the file name by combining them.

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
ArchiveFolder	No	The name of the folder where successfully processed mail will be archived in RFC822 format. If an archive folder is not defined, the adapter will not archive successfully processed mail and events will be deleted from the in progress Folder.	Default field is blank.	
DeliveryType	No	This property determines the order in which the events will be published. Supported values are either ORDERED (for delivery of events one at a time) or UNORDERED (to receive all of the events at the same time).	ORDERED	
DefaultObjectName	Yes	Default object that the adapter emits for inbound connections.	EmailBO	
EPDatabasePassword	No	The password used by the adapter event persistence feature. It is used to get a JDBC database connection from the data source.	None	
EPDatabaseSchemaName	No	The schema name of the database used by the adapter in the event of event persistence failure.	EmailEventSchema	

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
EPDatabaseUserName	No	The user name used by the adapter event persistence feature. It is used to get a JDBC database connection from the data source.	None	
EPDataSourceJNDIName	Yes	The JNDI name of the data source used by the event persistence class to get a JDBC database connection. The data source must be created in the application server.	None	
EPEventTableName	Yes	The name of the table that will be used by the adapter for event persistence. This value must be unique for each activation specification and the same table name should not be used by other instances of the same adapter or different adapters. If the table does not exist in the database, the adapter will create the table and use it.	EmailEventTable	

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
EventTypeFilter	No	Delimited list of event types that the adapter should deliver. The adapter will only deliver events that are listed. If the list is empty (null), no filter will be applied and all events will be delivered.	null	
FailedEventsFolder	No	The name of the folder where unsuccessfully processed mail or failed events will be archived in RFC822 format. If a failed events folder is not defined, the adapter will not archive unsuccessfully processed mail.	Default field is blank.	
filterFutureEvents	Yes	True or false value determines whether the adapter will inspect the timestamp on each event and compare them to the system time. If the event time is ahead of the system time, the event will not be delivered.	False	
Host	Yes	The host IP for the mail server	localhost	

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
InProgressFolder	Yes	The mail server folder where polled mails are first written to in RFC822 format. The name of the RFC822 format file corresponds to the message-ID of the mail. These details are retrieved by the adapter from the e-mail header associated with the mail.	Default field is blank.	
MatchAllCriteria	No	This field can be used to filter which mails are polled from the Poll folder. You may set AND conditions by separating them with a comma. You can also specify which mails you would not like to have polled by specifying NOT (!) in your search criteria.	Default registers as a blank field, meaning that you have not selected any filtering options.	You can use any of the following header fields when specifying search criteria: • content= • subject= • to= • cc= • from= • sender= • date=
MatchSomeCriteria	No	This field can be used to filter which mails are polled from the poll folder. You may set OR conditions by separating them with a comma. You can also specify which mails you would not like to have polled by specifying NOT (!) in your search criteria.	Default registers as a blank field, meaning that you have not selected any filtering options.	You can use any of the following header fields when specifying search criteria: • content= • subject= • to= • cc= • from= • sender= • date=

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
Password	Yes	The password associated with the username. This is used to authenticate the IMAP or POP3 session.	None	
PollFolders	Yes (POP3) No (IMAP)	The folder that you would like the adapter to poll.	For the POP3 protocol, this value is ignored and the property defaults to Inbox. Multiple poll folders can be defined if they are delimited by commas. For the IMAP protocol there is no default value.	
PollPeriod	Yes	This property is used by the WBI resource adapter base implementation. It determines the rate (in milliseconds) that the adapter will poll the event store for inbound events. The poll cycle is established at a fixed rate. If a polling cycle is delayed for any reason, the next cycle will immediately begin. If set to ″0″, the adapter will not wait between polling cycles.	500	

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
PollQuantity	Yes	This property is used by the WBI resource adapter base implementation and is used to determine the number of events to deliver to each endpoint per poll cycle. A positive integer greater than one is required.	1	
Protocol	Yes	The protocol to be used for inbound communication with the mail server. This field can be set to either IMAP or POP3	POP3	
RetryInterval	No	The value set for this property determines the amount of time (in milliseconds) that the adapter will continually attempt to reconnect with the EIS in the event of an error related to the inbound EIS connection.	60000	

Table 11. Activation specification properties (continued)

Property	Required	Description	Default	Example
useInboundTransactions	Yes	If set to true, the adapter will attempt to store XID information in the event store and the adapter will provide assured once event delivery. If set to false, the adapter will not guarantee that the mail will be sent once and only once, but the performance is improved. **Note:** This property is only used if the mail database is transactional. If it is not, the transaction can be used, regardless of the configuration.	True	
Username	Yes	The user ID associated with the IMAP or POP3 session	None	

Settings for controlling bidirectional transformation

Within each category of adapter properties, certain properties can be set to control bidirectional transformation of content or metadata. Properties controlling bidirectional transformation can be set for the resource adapter, the managed connection factory, and the activation specification; data transformation properties can also be set to control bidirectional transformation.

Resource adapter properties

The following resource adapter properties can be set to control bidirectional transformation.

- EIS BiDi Format
- Metadata BiDi Format
- Skip BiDi Transformation
- EIS BiDi Special Format

Managed (J2C) connection factory properties

The following managed (J2C) connection properties can be set to control bidirectional transformation.

- Username
- Password

Activation specification properties

The following activation specification properties can be set to control bidirectional transformation.

- PollFolders
- InProgressFolder
- ArchiveFolder
- FailedEventsFolder
- ArchiveFileName
- All event persistence-related properties
- EPDataSourceJNDIName

Business object properties

The properties in this section should be configured using the enterprise service discovery wizard before deployment or with the WebSphere Process Server administrative console after deployment.

EmailBO business object properties

The EmailBO business object is generated during enterprise service discovery. It includes all of the details required by the adapter for inbound and outbound communications.

Table 12. EmailBO business object properties

Attribute name	Type	Required	Globalized	Default	Description
headerList	HeaderBO[]	Yes	Yes	No default value	Will contain details of all the headers on the polled e-mail

Table 12. EmailBO business object properties (continued)

Attribute name	Type	Required	Globalized	Default	Description
Encoding	String	no	no	No default value	For outbound, encoding is used for pass-through data encoding. For encoding headers, the adapter uses the charset value provided by the annotation of mailContent specific business object (Customer). In cases where both the encoding at business object and charset at annotation of mailContent specific business object are present, the charset at the annotation level of the mailContent specific business object takes precedence over the top level encoding value at EmailBO. Headers should always be encoded the same way as mailContent. The adapter also uses encoding to encode headers in the absence of a charset (specific business object) or encoding (pass through) value at mailContent. In the absence of charset or encoding value at any level, the adapter uses the encoding value at the top level to encode that specific level content.

Table 12. EmailBO business object properties (continued)

Attribute name	Type	Required	Globalized	Default	Description
mailContent	anyType	Yes	Yes	No default value	For inbound events, this stores the content/data of the e-mail read and passed into the server by the adapter. For outbound events, this contains content data but not the e-mail attachment. In a pass-through scenario, it contains an unstructured content business object.
mailAttachments	AttachmentBO[]	No	Yes	No default value	Will contain content details for all the attachments of the e-mail.
fileReferences	String[]	No	Yes	No default value	Will contain a list of files that needs to be attached to the e-mail. During outbound, the J2EE client specifies absolute paths of the files in this field. The adapter reads those files from the local file system (where the adapter is running) and attaches them to the email that is created. The property is not used during inbound. Note: There is only support for attaching local files in this release.

HeaderBO business object properties

HeaderBO business object properties are used to store all standard (RFC822) e-mail headers. Customized headers and their related information will be tracked by the

adapter with the help of the headerList entries in the EmailBO object. These properties are filled either by the adapter or by the requesting J2EE client at runtime.

Table 13. Standard headers supported by the Email adapter

Header name	Required	Description
From	Yes	The identity of the person or persons sending the e-mail.
To	Yes	The identity of the primary e-mail recipient.
Cc	No	The identity of any person or persons to be copied on the e-mail.
Bcc	No	The identity of any person or persons to be copied on the message, but not identified to any of the other recipients.
Date	No	The date the e-mail was created.
Subject	No	Summary of the message.
Message-ID	No	This field contains a unique identifier (the local-part address unit) which refers to the specific version of the sent message. The uniqueness of the message identifier is guaranteed by the host that generated it. It is machine readable but not necessarily meaningful to humans. A message identifier pertains to exactly one instantiation of a particular message; subsequent revisions to the message each receive new message identifiers.
Resent-Message-ID	No	This field contains a unique identifier (the local-part address unit) which refers to the specific version of the sent message. The uniqueness of the message identifier is guaranteed by the host that generated it. It is machine readable but not necessarily meaningful to humans. A message identifier pertains to exactly one instantiation of a particular message; subsequent revisions to the message each receive new message identifiers.

Table 13. Standard headers supported by the Email adapter (continued)

Header name	Required	Description
Sender	No	The authenticated identity of the agent (person, system, or process) sending the message.
Reply-To	No	Indicates which mailbox or mailboxes the reply messages will to be sent to.
In-Reply-To	No	Identifies previous correspondence which this message answers. **Note:** If message identifiers are used in this field, they must use the msg-id specification format.
Resent-To	No	The identity of the primary recipients of the e-mail.
Resent-Cc	No	The identity of the secondary (informational) recipients of the e-mail.
Resent-Bcc	No	The identity of any additional recipients of the message. The contents of this field are not included in the copies of the message sent to the primary and secondary recipients. Some systems may choose to include the text of the ″Bcc″ field only in the author's copy, while others may also include it in the text sent to all those indicated in the Bcc list.
Resent-Date	No	The date the e-mail was forwarded.
Resent-From	No	The person or persons the message was sent from.
Resent-Sender	No	The authenticated identity of the agent (person, system, or process) who sent the message.
Resent-Reply-To	No	Indicates which mailbox(es) you would like reply messages to be sent to.
References	No	The contents of this field identify other correspondence that the message references. **Note:** If message identifiers are used, they must use the msg-id specification format.
Keywords	No	Any keywords or phrases, separated by commas.

Table 13. Standard headers supported by the Email adapter (continued)

Header name	Required	Description
Comments	No	Permits adding text comments onto the message without disturbing the contents of the message's body.
Encrypted	No	This field is used to note whether you would like keep the contents private. Encrypted=true if you would like to use this feature.

MailAttachmentBO business object properties

The MailAttachmentBO stores the business objects that correspond with attachments in an e-mail. MailAttachmentBO properties determine how the attachment is stored in a business object or attached in an email. These properties are defined either by the adapter or by the requesting J2EE client at runtime.

MailAttachmentBO business object property values

Attribute name	Type	Required	Globalized	Default	Description
attachmentName	String	Yes	Yes	No default value	Specifies the name of the attachment
appBOData	anyType	Yes	Yes	No default value	The mail attachment content. anyType datatype can hold hexBinary or any SDO type of data. The data binding would decipher the hexBinary content as the unstructured content BO.

Business object attributes

Business object attributes store important Name, Type, and Key information about the business object. They are set during enterprise service discovery and can be reset in the WebSphere Process Server administrative console.

Table 14. Business object attribute values

Attribute name	Description
Name	Indicates the name of the business object attribute
Type	Indicates the Type of the business object attribute. The types can be simple or complex and include Boolean, String, Integer, Byte[], HashMap, and ArrayList
Key	The Email adapter uses RFC-822 format and uses the Message-ID property value as the unique identifier for e-mails. Foreign keys are not supported.

Sample artifacts supplied for inbound tutorials

The following artifacts have been shipped with the product to create inbound events in the tutorials. The events that are created by the sample bat files for pass-through scenario testing will be different from the ones created for non-pass-through scenario tests. The mail format is the same, but the content and attachments will differ.

Table 15. Sample artifacts supplied for inbound tutorials

Artifact name	Location	Description
WbiAddress.xsd	Samples\XSDs.zip and Samples\XSDs.tar	Provides the data description for the WbiAddress business object.
WbiPhone.xsd	Samples\XSDs.zip and Samples\XSDs.tar	Provides the data description for the WbiPhone business object. It can be used as its own type or as a child type of the WbiAddress business object.
EmailEventCreator.class	Samples\ EmailEventCreator.zip or Samples\ EmailEventCreator.tar	Class file of the Java program used to create events for the Email adapter.
MyAddress.xml	Samples\ EmailEventCreator.zip or Samples\ EmailEventCreator.tar	Example of an XML file corresponding to WbiAddress type, with appropriate data filled in. This file will be attached to the mail and sent out during event creation.
createEvents.bat	Samples\ EmailEventCreator.zip	A batch file that invokes the Java program with respective command line parameters used to help users create events. **Note:** All the command line parameters need to be populated by the user before it is used.
createEvents_sample_ passthrough.bat	Samples\ EmailEventCreator.zip	A batch file that invokes the EventCreator program with specific command line parameters to help users create events for pass-through mode.
createEvents_sample_ nonpassthrough.bat	Samples\ EmailEventCreator.zip	A batch file that invokes the EventCreator program with specific command line parameters to help users create events for non-pass-through mode.
createEvents.sh	Samples\ EmailEventCreator.tar	Shell script corresponding to the createEvents.bat file.
createEvents_sample_ passthrough.sh	Samples\ EmailEventCreator.tar	Shell script corresponding to the createEvents_sample_ passthrough.bat file.

Table 15. Sample artifacts supplied for inbound tutorials (continued)

Artifact name	Location	Description
createEvents_sample_ nonpassthrough.sh	Samples\ EmailEventCreator.tar	Shell script corresponding to the createEvents_sample_ nonpassthrough.bat file.

Adding jar files to WebSphere Integration Developer versions 6.0.1.1 and earlier

If you are using WebSphere Integration Developer version 6.0.1.1 or earlier, you must manually add three jar files to the classpath of the connector project.

You must have installed the adapter and all of the adapter prerequisites before the jar files can be added to the connector project in WebSphere Integration Developer.

1. Open WebSphere Integration Developer.
2. In J2EE perspective, right-click the connector project and select **Properties**.
3. Select **Java Build Path** and click **Add External Jars**.
4. Select your WebSphere Process Server or Enterprise Server Bus Install/lib folder and select ffdcSupport.jar, aspectjrt.jar and icu4j_3_2.jar.
5. Click **Open** and then **OK**.

Messages

The messages issued by IBM WebSphere Adapters are documented in the WebSphere Adapters, version 6.0.2 information center.

You can view the adapter messages at the following link: WebSphere Adapters messages..

Related product information

The following links, information centers, Redbooks, and Web pages contain related information for the IBM WebSphere Adapter for Email.

Additional information you might need

Table 16. WebSphere Adapters information you might need

Information	How to find it
How to edit business objects using the Business Object Editor	In the IBM WebSphere Business Process Management information center, which includes documentation for WebSphere Integration Developer, search for the topic, ″Editing Business Objects.″
How to uninstall a deployed adapter	On the WebSphere Application Server library page, open the information center for your version of WebSphere Application Server and search for the topic, ″Uninstalling applications.″

Information for related products

- WebSphere Adapters, Version 6.0
- WebSphere Business Integration Adapters
- WebSphere Integration Developer
- WebSphere Process Server

- WebSphere Enterprise Service Bus
- WebSphere Application Server

Redbooks

- WebSphere Adapter Development Redbook
- WebSphere Redbooks domain

developerWorks® resources

- WebSphere Adapter Toolkit
- WebSphere business integration zone

Support and assistance

- WebSphere Adapters product support
- WebSphere Adapters technotes - in the **Additional search terms** field, specify the name of the adapter and click **Go**.

Chapter 14. Glossary

The glossary of terms for IBM WebSphere Adapters is included in the WebSphere Adapters, version 6.0.2 information center.

You can view it at the following link: WebSphere Adapters glossary.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing 2-31 Roppongi 3-chome, Minato-ku Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation 577 Airport Blvd., Suite 800 Burlingame, CA 94010 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows: © (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

IBM and related trademarks: http://www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.

This product includes software developed by the Eclipse Project (http://www.eclipse.org/).

Index

T

U

V

W

X

Printed in USA