IBM WebSphere Adapters
Version 7 Release 5 Fix Pack 3 (7.5.0.3)

IBM WebSphere Adapter for PeopleSoft Enterprise User Guide
Version 7 Release 5 Fix Pack 3
(7.5.0.3)

IBM

IBM WebSphere Adapters
Version 7 Release 5 Fix Pack 3 (7.5.0.3)

*IBM WebSphere Adapter for PeopleSoft
Enterprise User Guide
Version 7 Release 5 Fix Pack 3
(7.5.0.3)*

IBM

Contents

Chapter 1. Overview of WebSphere Adapter for PeopleSoft Enterprise

With WebSphere® Adapter for PeopleSoft Enterprise, you can create integrated processes that exchange information with PeopleSoft Enterprise through a standard interface. This interface shields the client application from analyzing lower-level details about implementation of the application or data structures used on the PeopleSoft Enterprise server.

With the adapter, a client application can send and receive requests from a PeopleSoft Enterprise server. For example, the application can send a request to query an employee record in an HR table, or it can receive events such as notification about an updated employee record from the server.

To make it possible to communicate in a standard way, WebSphere Adapter for PeopleSoft Enterprise complies with the Java™ Platform, Enterprise Edition (JEE) Connector Architecture (JCA). JCA standardizes the way application components, application servers, and enterprise information systems, such as PeopleSoft Enterprise server, interact with each other. WebSphere Adapter for PeopleSoft Enterprise enables JCA-compliant application servers such as IBM® Business Process Manager or WebSphere Enterprise Service Bus to connect to and interact with the PeopleSoft Enterprise server. Clients running on the JCA-compliant server can then communicate with the PeopleSoft Enterprise server in a standard way using business object.

The adapter can run in JCA-compliant environments such as IBM Business Process Manager or WebSphere Enterprise Service Bus, WebSphere Application Server, and WebSphere Message Broker. The way you set up and use the adapter depends upon the environment in which the adapter runs.

For example, consider that you are setting up an adapter using IBM Integration Designer tooling on IBM Business Process Manager or WebSphere Enterprise Service Bus, and your company uses PeopleSoft Enterprise to coordinate most of its business operations. PeopleSoft Enterprise includes a business function that returns a list of employees in response to a range of skill codes. A client application might be able to use this function as part of an overall business process. For example, the technical support department within the company dispatches employees to customer sites to troubleshoot and repair equipment and, as part of that process, must obtain a list of employees with certain skills within a defined geographic region.

To use this function, however, the client application that the technical support department uses must understand the low-level implementation details and data structures of the PeopleSoft function. Information technology resources and time must link the client application and the PeopleSoft function.

With WebSphere Adapter for PeopleSoft Enterprise, you can automatically generate an interface to the PeopleSoft function to hide the lower-level details of the function. Depending on your plans for using the adapter, you can either embed it with the deployed application or install it as a stand-alone component that more than one application uses.

The embedded adapter is part of an application that is deployed to IBM Business Process Manager or WebSphere Enterprise Service Bus. The application component interacts with the adapter instead of interacting with the PeopleSoft function.

The component that you generate with the external service wizard of IBM Integration Designer uses a standard interface and standard business objects, taking the business object sent by the client application, and calling the PeopleSoft function. It then returns a business object to the client application.

The client application does not deal directly with the PeopleSoft function. Instead, the component calls the function and returns the results.

In the example, the client application that needed the list of employees sends a business object with the range of skill codes to the PeopleSoft adapter component. The client application, in return, receives the results (the list of employees) in the form of a standard business object. The client application does not need information about the way the function worked or the structure of the data. The adapter component performs all interactions with the PeopleSoft function.

Similarly, the client application might be alerted about a change to the data on the PeopleSoft Enterprise server (for example, a change to the skills set of a specific employee). You can generate an adapter component that polls for such events on the PeopleSoft Enterprise server and notifies client applications with the update.

Hardware and software requirements

The hardware and software requirements for WebSphere Adapters are provided on the IBM Support website.

To view hardware and software requirements for WebSphere Adapters, see http://www.ibm.com/support/docview.wss?uid=swg27006249.

Additional information

The following links provide additional information you might need to configure and deploy your adapter:
- The compatibility matrix for WebSphere Business Integration Adapters and WebSphere Adapters identifies the supported versions of required software for your adapter. To view this document, go to the WebSphere Adapters support page: http://www.ibm.com/support/docview.wss?uid=swg27006249.
- Technotes for WebSphere Adapters provide workaround and additional information that are not included in the product documentation. To view the technotes for your adapter, go to the following Web page, select the name of your adapter from the **Product category** list, and click the search icon: http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8 &dc=DB520+D800+D900+DA900+DA800+DB560&dtm.

Technical overview of WebSphere Adapter for PeopleSoft Enterprise

The adapter supports the exchange of business data between the PeopleSoft Enterprise server and IBM Business Process Manager or WebSphere Enterprise Service Bus by connecting to two layers of PeopleTools application programming interface classes that reveal the underlying business data for integration.

The adapter establishes bidirectional connectivity with PeopleSoft Enterprise server by connecting to two layers of PeopleTools application programming interfaces:

1. The adapter accesses the primary API layer to create a session instance and to connect to the application server through the Jolt port.
2. The adapter then accesses the PeopleSoft Component Interface API, which reveals underlying business objects, logic, and functionality.

In PeopleSoft, a component is a set of pages grouped together for a business purpose (such as an employee profile), and a component interface is an API that provides access to a component from an external application. After the adapter connects to the component interface, the following entities are exposed to the adapter and available for integration:

- All business objects in the component interface definition.
- PeopleCode methods associated with the underlying components.
- Records, except searches and menu-specific processing options.

Outbound processing

WebSphere Adapter for PeopleSoft Enterprise supports synchronous outbound request processing, so that when the client application sends a request in the form of a business object to the adapter, the adapter processes the request, and returns a business object that represents the result of the operation to the client application.

When the adapter receives a WebSphere business object hierarchy, the adapter processes it as follows:

1. The adapter extracts metadata that identifies the appropriate PeopleSoft component interface to access.
2. The adapter extracts the outbound operation to perform.
3. After accessing the component interface, the adapter sets the keys from values specified in the business objects. If key values are not generated, for example with a Create operation, the PeopleSoft application generates key fields.
4. After retrieving the PeopleSoft objects, the adapter instantiates an existing component interface to delete, retrieve, update, or create a component interface.
5. If update operations (Create, Update) are to be performed, the adapter populates the component interface with data from the WebSphere business object hierarchy. If Delete operations are required, the adapter populates the component interface only with StatusColumnName and value information.

Note: For Create and Update operations only, the adapter processes attributes in the order defined in the business object. For example, if a complex attribute is between two simple attributes, the adapter processes the simple attribute at the first position, then the complex attribute, and then the simple attribute. After the changes are made, the component interface is saved to commit the data to the PeopleSoft database.

Supported outbound operations

WebSphere Adapter for PeopleSoft Enterprise supports the following outbound operations:

- Apply Changes
- Create
- Delete
- Exists

- Retrieve
- Retrieve All
- Update

The following table describes each of these operations.

Table 1. Supported outbound operations

Operation	Description
ApplyChanges	Updates, creates, or deletes the business object based on the delta information contained in the change summary.
Create	Creates the business object.
Delete	Deletes the business object and its children. Because the adapter supports only logical deletes, objects are marked as deleted but not removed.
Exists	Checks for the existence of incoming business objects.
Retrieve	Retrieves the PeopleSoft component and maps component data onto the business object.
RetrieveAll	Retrieves multiple instances of the PeopleSoft component and maps component data onto the business object. The following errors can result from this operation: RecordNotFoundFault– This fault is generated when the input object does not exist in the enterprise information system and if the managed connection factory property for the "Show error on empty result set (showErrorOnEmptyResultSet)" on page 130 property is set to True and the "Use Invalid Request Fault (useInvalidRequestFault)" on page 132 is set to False. To get an empty business object without the RecordNotFoundFault, you can either modify the .import file and set the "Show error on empty result set (showErrorOnEmptyResultSet)" on page 130 property to False, or after you deploy the outbound application on IBM Business Process Manager, configure the Managed Connection Factory property "Show error on empty result set (showErrorOnEmptyResultSet)" on page 130 to False. MatchesExceededLimitFault – This fault is generated when the number of matching records in the database exceeds the value of the "Maximum number of records for RetrieveAll operation (MaxRecords) " on page 133 property that is defined in the interaction specification. The MatchCount attribute of the fault contains the actual number of matches that the adapter found in the database, so that you can either increase the limit or refine the search. Note: If the "Maximum number of records for RetrieveAll operation (MaxRecords) " on page 133 property is set to a large number, problems can occur due to a lack of sufficient memory, depending upon the size and number of business objects returned.
Update	Updates the corresponding PeopleSoft component with the incoming business object.

Inbound processing

WebSphere Adapter for PeopleSoft Enterprise supports asynchronous inbound processing. This means that the adapter polls the PeopleSoft Enterprise server at specified intervals for events. When the adapter detects an event, it converts the event data into a business object and sends it to the client application.

To use inbound processing, you must create a custom event project in PeopleSoft Enterprise server.

Event store

The event store is a table that holds events that represent data changes until the polling adapter can process them. The adapter uses the event store to keep track of event entities.

To use inbound processing, you must use PeopleTools Application Designer to create a custom project for event notification. The custom project uses two PeopleCode functions that determine the way future events are processed, and the custom project creates the event store the adapter needs for inbound processing. Each time a business object is created, updated, or deleted, the PeopleCode function used in the project and then added to the component interface inserts a new record in the event store, with the appropriate object name, keys, and status value.

During inbound processing, the adapter polls the events from the event store at configured poll intervals. In each poll call, a configured number of events are processed by the adapter. The order of event processing is based on the ascending order of priority and the ascending order of the event time stamp. The events with the Ready for poll (0) status are picked up for polling in each poll cycle. The adapter uses the object name and object key to retrieve the corresponding business object.

To ensure that an event is delivered only once to the target application, set the activation specification property AssuredOnceDelivery to `true`. On setting this to true, an XID (transaction ID) value is set for each event in the event store. After an event is obtained for processing, the XID value for that event is updated in the event store. The event is then delivered to its corresponding export, and its status is updated to show that event delivery was completed. If the application is stopped before the event can be delivered to the export or if the delivery fails, the event might not be processed completely. In this case, the XID value represents an in-progress status, and the XID column ensures that the event is reprocessed and sent to the export. After the database connection is re-established or the adapter starts again, the adapter checks for events in the event table that have a value in the XID column of Ready for Poll (0). The adapter processes these events first, and then polls the other events during the poll cycles.

The adapter uses special handing for events that have status code (99) to indicate that those events will occur in the future. During a poll cycle, when the adapter retrieves events with a future status, it compares the system time with the time stamp on each event. If the timestamp on the event is earlier than or equal to the system time, the adapter processes the event and changes the event status to Ready for Poll (0).

Note: If you want the adapter to process events with a future status in the present time, use the function `IBM_PUBLISH_EVENT` instead of `IBM_FUTURE_PUBLISH_EVENT`. Doing so means that the event is identified as Ready for Poll (0) instead of Future (99).

As events are retrieved from the event store and processed, the status of the event changes to reflect its state, as shown in the following table.

Table 2. Event status values

Status short name	Description	Event table value
Error processing event	An error occurred during event processing.	-1
Ready for poll	The event is ready to be picked up.	0
Success	The event is delivered to the event manager.	1
Deleted	The event is processed successfully and is removed from the event store.	4
Future Events	These events that are processed at a future date.	99

Business objects

To send data or to obtain data from the PeopleSoft Enterprise, the WebSphere Adapter for PeopleSoft Enterprise uses business objects. A business object is a structure that consists of data, the action to be performed on the data, and additional instructions, if any, for processing the data. The data can represent either a business entity, such as an invoice or an employee record, or unstructured text.

How business objects are created

Use the external service wizard, which is started from IBM Integration Designer, to create business objects. The wizard connects to the application, discovers data structures in the application, and generates business objects to represent them. It also generates other artifacts needed by the adapter.

Business object structure

The adapter supports business objects that are hierarchically structured. The top-level business object must have a one-to-one correspondence with the PeopleSoft component interface, and collections that occur within the top-level object are children of it. Information that the adapter uses to process the business object is stored in the application-specific information for the object and in each of its attributes.

The following table describes the attributes that compose a business object.

Attribute property	Description
Name	Name of the business object attribute.

Attribute property	Description
Type	Type of the business object attribute. The adapter uses character mapping between PeopleSoft component property types and the generated business object attribute types. PeopleSoft component property types map to generated attribute types in the following manner: • CHAR maps to attribute type String. • NUMBER maps to attribute type Integer. • SIGNED NUMBER maps to attribute type Integer. • DATE maps to attribute type String. • TIME maps to attribute type String. • DTTM maps to attribute type String.
Key	Application-specific information for the primary key. Child business objects also inherit keys from their parent business object.
Cardinality	Single cardinality for simple attributes; multiple cardinality for container attributes.

Business graphs

You can optionally choose, during adapter configuration, to generate a business graph. In version 6.0.2, each top-level business object is contained in a business graph, which includes a verb that an application can use to specify additional information about the operation to be performed. Beginning version 7.0, business graphs are required only in these situations:

• If you need to use the outbound ApplyChanges operation
• When adding business objects to a module created with an earlier version

If business graphs exist, they are processed, but the verb is ignored for all operations except ApplyChanges.

The external service wizard

The external service wizard is a tool you use to configure your adapter before deploying it to IBM Business Process Manager or WebSphere Enterprise Service Bus. The external service wizard establishes a connection to the PeopleSoft Enterprise server, discovers business objects and services (based on search criteria you provide), and generates business objects based on the services discovered.

Using IBM Integration Designer, you establish a connection to the PeopleSoft Enterprise server to browse the metadata repository on the PeopleSoft Enterprise server.

In the wizard, you select the business objects you need. The wizard automatically generates the XSD schemas that represent the business objects and the corresponding Java data bindings.

The result of running the external service wizard is a module that contains the business objects and services along with the adapter. You deploy this module on IBM Business Process Manager or WebSphere Enterprise Service Bus.

Log and Trace Analyzer

The adapter creates log and trace files that can be viewed with the Log and Trace Analyzer.

The Log and Trace Analyzer can filter log and trace files to isolate the messages and trace information for the adapter. It can also highlight the messages of an adapter and trace information in the log viewer.

The adapter's component ID for filtering and highlighting is a string composed of the characters PSFTRA plus the value of the adapter ID property. For example, if the adapter ID property is set to 001, the component ID is PSFTRA001.

If you run multiple instances of the same adapter, ensure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate the log and trace information to a particular adapter instance. By making the first seven characters of an adapter ID property unique, the component ID for multiple instances of that adapter is also unique, allowing you to correlate the log and trace information to a particular instance of an adapter. For example, when you set the adapter ID property of two instances of WebSphere Adapter for PeopleSoft Enterprise to 001 and 002. The component IDs for those instances, PSFTRA001 and PSFTRA002, are short enough to remain unique, enabling you to distinguish them as separate adapter instances. However, instances with longer adapter ID properties cannot be distinguished from each other. If you set the adapter ID properties of two instances to Instance01 and Instance02, you will not be able to examine the log and trace information for each adapter instance because the component ID for both instances is truncated to PSFTRAInstanc.

For outbound processing, the adapter ID property is located in both the resource adapter and managed connection factory property groups. If you update the adapter ID property after using the external service wizard to configure the adapter for outbound processing, be sure to set the resource adapter and managed connection factory properties consistently. It prevents inconsistent marking of the log and trace entries. For inbound processing, the adapter ID property is located only in the resource adapter properties, so this consideration does not apply.

For more information, see the "Adapter ID (adapterID)" on page 123 property.
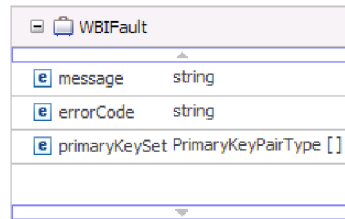
Business faults

The adapter supports business faults, which are exceptions that are anticipated and declared in the outbound service description, or import. Business faults occur at predictable points in a business process, and are caused by a business rule violation or a constraint violation.

Although IBM Business Process Manager and WebSphere Enterprise Service Bus support other types of faults, the adapter generates only business faults, which are referred to as *faults* in this documentation. Not all exceptions become faults. Faults are generated for errors that are actionable, that is, errors that can have a recovery action that does not require the termination of the application. For example, the adapter generates a fault when it receives a business object for outbound processing that does not contain the required data or when the adapter encounters certain errors during outbound processing.

Fault business objects

The external service wizard creates a business object for each fault that the adapter can generate. In addition, the wizard creates a WBIFault superset business object, which has information common to all faults, such as the message, errorCode, and primaryKeySet attributes, as shown in the figure.

Figure 1. The structure of the WBIFault business object

Some faults contain the matchCount attribute, to provide additional information about the error. For others, WBIFault contains all the information needed to handle the fault. For information about business faults, see "Fault business objects listing" on page 113.

Chapter 2. Planning for adapter implementation

To implement the IBM WebSphere Adapter for PeopleSoft Enterprise, you must plan for inbound and outbound processing and consider security and performance requirements.

Before you begin

Before you begin to set up and use the adapter, you need to possess a thorough understanding of business integration concepts, the capabilities, and requirements of the integration development tools and runtime environment you use, and the PeopleSoft Enterprise environment where you build and use the solution.

To configure and deploy WebSphere Adapter for PeopleSoft Enterprise, you need to understand and have experience with the following concepts, tools, and tasks:

*   The business requirements of the solution you are building.
*   Business integration concepts and models, including the Service Component Architecture (SCA) programming model.
*   The capabilities provided by the development tools you use to build the solution. You need to know how to use these tools to create modules, test components, and complete other integration tasks.
*   The capabilities and requirements of the runtime environment used for the integration solution. You need to know how to configure and administer the host server and how to use the administrative console to set and modify property definitions, configure connections, and manage events.
*   PeopleSoft integration architecture, including the relationship between component interfaces and underlying components.
*   The PeopleTools version that supports your applications. You need to be able to use PeopleSoft Application Designer to access the object definitions that compose a business application, and to create and add projects to components.
*   The programming environment required by PeopleSoft (PeopleCode). You need to understand the role of PeopleCode within components, and know how to add PeopleCode to the projects you create in PeopleSoft Application Designer.

Security

The adapter uses the J2C authentication data entry, or the authentication alias feature of Java EE security to provide secure user name and password authentication. For more information about security features, see the documentation for IBM Business Process Manager or WebSphere Enterprise Service Bus.

Support for protecting sensitive user data in log and trace files

You can configure the adapter to prevent sensitive or confidential data, in the log and trace files, from being viewed by users without authorization.

Log and trace files for the adapter can contain data from your PeopleSoft Enterprise server, which might contain sensitive or confidential information.

Sometimes these files might be seen by individuals without authorization to view sensitive data. For example, a support specialist must use the log and trace files to troubleshoot a problem.

To protect the data in such situations, the adapter lets you specify whether you want to prevent confidential user data from displaying in the adapter log and trace files. You can select this option in the external service wizard or change the HideConfidentialTrace property. When this property is enabled, the adapter replaces the sensitive data with XXX's.

See "Managed connection factory properties" on page 127 for information about this optional property.

The following types of information are considered potentially sensitive data and are disguised:
* The contents of a business object
* The contents of the object key of the event record
* User name and password
* The URL used to connect to the PeopleSoft Enterprise server

The following types of information are not considered user data and are not hidden:
* The contents of the event record that are not part of the event record object key, for example, the XID, event ID, business object name, and event status
* Business object schemas
* Transaction IDs
* Call sequences

User authentication

The adapter supports several methods for supplying the user name and password that are needed to connect to the PeopleSoft Enterprise server. By understanding the features and limitations of each method, you can pick a method that provides the appropriate level of security and convenience for your application.

To integrate an adapter into your application, a user name and password are needed at the following times:
* When the external service wizard connects to PeopleSoft Enterprise server to extract, or *discover*, information about the objects and services that you can access with the adapter.
* At run time on IBM Business Process Manager or WebSphere Enterprise Service Bus, when the adapter connects to PeopleSoft Enterprise server to process outbound requests and inbound events.

Authentication in the wizard

The external service wizard prompts for the connection information for the discovery process, and then reuses it as the default values of the adapter properties that specify the connection information used at run time. You can use a different user name and password while running the wizard than you use when the application is deployed to the server. You can even connect to a different PeopleSoft Enterprise server, although the schema name must be the same in both databases. For example, while developing and integrating an application that uses

WebSphere Adapter for PeopleSoft Enterprise, you might not use the production database; using a test database with the same data format but fewer, simulated records allows you to develop and integrate the application without impacting the performance of a production database and without encountering restrictions caused by the privacy requirements for customer data.

The wizard uses the user name and password that you specify for the discovery process only during the discovery process; they are not accessible after the wizard is completed.

Authentication at run time

At run time, the adapter needs to provide the user name and password to connect to the PeopleSoft Enterprise server. To connect without user intervention, the adapter must access a saved copy of the user information. In a server environment, there are several methods for saving user information. You can configure the adapter to get your user information, through any of the following methods:

- Adapter properties
- J2C authentication alias

Saving the user name and password in adapter properties is a direct way to provide this information at run time. You provide this user name and password when you use the external service wizard to configure your module. Although directly specifying the user name and password seems the most straightforward method, it has important limitations. Adapter properties are not encrypted; the password is stored as clear text in fields that are accessible to others on the server. Also, when the password changes, you must update the password in all instances of the adapter that access that PeopleSoft Enterprise server. This includes the adapters embedded in application EAR files and adapters that are separately installed on the server.

Using a J2C authentication data entry, or authentication alias, created with the Java Authentication and Authorization Service (JAAS) feature of Java EE security is a robust, secure way to deploy applications. An administrator creates the authentication alias that is used by one or more applications that need to access a system. The user name and password must be known only to that administrator, who can change the password in a single place, when a change is required.

Deployment options

There are two ways to deploy the adapter. You can either embed it as part of the deployed application, or you can deploy it as a stand-alone RAR file. The requirements of your environment affect the type of deployment option you choose.

The following are the deployment options:

- **With module for use by single application**: With the adapter files embedded in the module, you can deploy the module to any application server. Use an embedded adapter when you have a single module using the adapter or if multiple modules need to run different versions of the adapter. Using an embedded adapter enables you to upgrade the adapter in a single module without the risk of destabilizing other modules by changing their adapter version.
- **On server for use by multiple applications**: If you do not include the adapter files in a module, you must install them as a stand-alone adapter on each

application server where you want to run the module. Use a stand-alone adapter when multiple modules can use the same version of the adapter and you want to administer the adapter in a central location. A stand-alone adapter can also reduce the resources required by running a single adapter instance for multiple modules.

An embedded adapter is bundled within an enterprise archive (EAR) file and is available only to the application with which it is packaged and deployed.
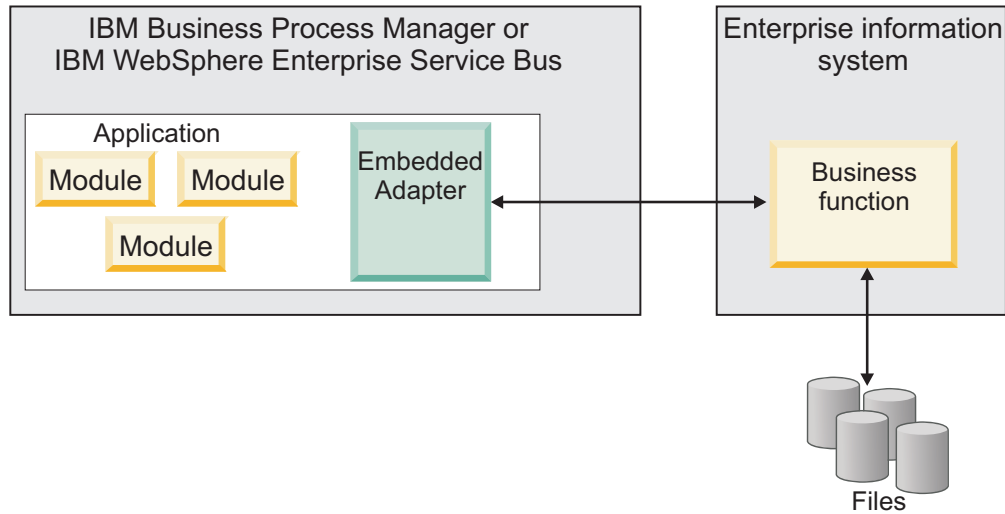
A stand-alone adapter is represented by a stand-alone resource adapter archive (RAR) file, and when deployed, it is available to all deployed applications in the server instance.
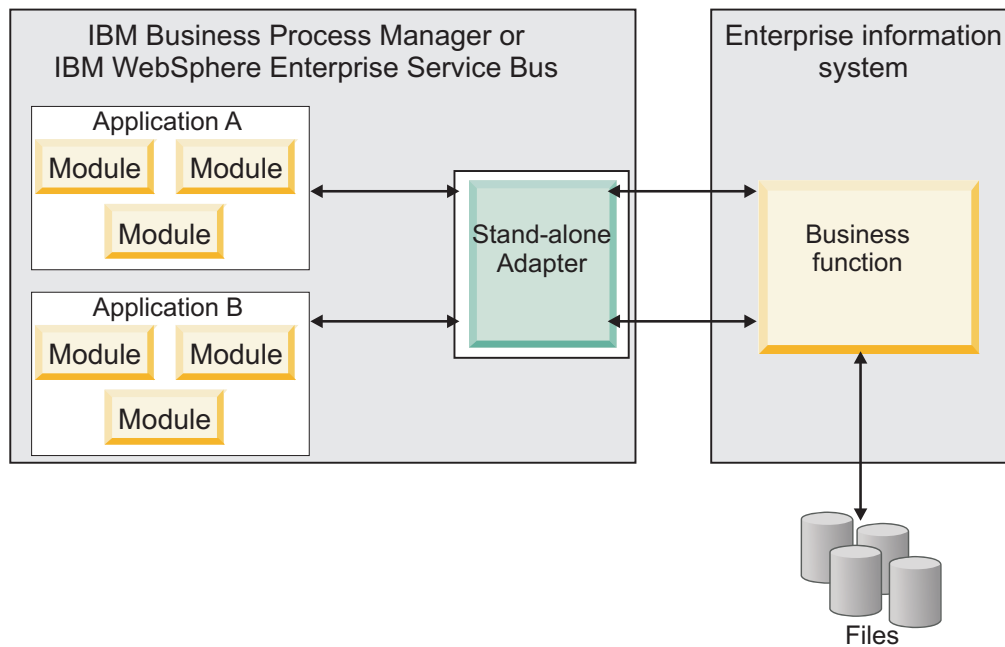
While creating the project for your application using IBM Integration Designer, you can choose how to package the adapter [either bundled with the (EAR) file or as a stand-alone (RAR) file]. Your choice affects how the adapter is used in the run time environment and how the properties for the adapter are displayed on the administrative console.

Choosing either to embed an adapter with your application or to deploy the adapter as a stand-alone module depends on how you want to administer the adapter. If you want a single copy of the adapter and do not care about disruption to multiple applications when you upgrade the adapter, then you would be more likely to deploy the adapter as a stand-alone module.

If you plan to run multiple versions, and if you care more about potential disruption when you upgrade the adapter, you would be more likely to embed the adapter with the application. Embedding the adapter with the application allows you to associate an adapter version with an application version and administer it as a single module.

Considerations for embedding an adapter in the application

Consider the following items if you plan to embed the adapter with your application:

- An embedded adapter has class loader isolation.

  A class loader affects the packaging of applications and the behavior of packaged applications deployed on run time environments. *Class loader isolation* means that the adapter cannot load classes from another application or module. Class loader isolation prevents two similarly named classes in different applications from interfering with each other.

- Each application in which the adapter is embedded must be administered separately.

Considerations for using a stand-alone adapter

Consider the following items if you plan to use a stand-alone adapter:

- Stand-alone adapters have no class loader isolation.

  Because stand-alone adapters have no class loader isolation, only one version of any defined Java artifact is run and the version and sequence of that artifact is undetermined. For example, when you use a stand-alone adapter there is only *one* resource adapter version, *one* adapter foundation class (AFC) version, or *one* third-party JAR version. All adapters deployed as stand-alone adapters share a single AFC version, and all instances of a defined adapter share the code version. All adapter instances using a given third-party library must share that library.

- If you update any of these shared artifacts, all applications using the artifacts are affected.

  For instance, if you have an adapter that is working with server version X, and you update the version of the client application to version Y, your original application might stop working.

- Adapter Foundation Classes (AFC) is compatible with previous versions, but the latest AFC version must be in every RAR file that is deployed in a stand-alone manner.

  If more than one copy of any JAR file is in the class path in a stand-alone adapter, the one that is used is random; therefore, they all must be the latest version.

Considerations while deploying adapters with different versions

When you install multiple adapters with different versions of CWYBS_AdapterFoundation.jar, and if a lower version of the CWYBS_AdapterFoundation.jar is loaded during run time, the adapter returns the

ResourceAdapterInternalException error message, due to a version conflict. For example, when you install Oracle E-Business Suite adapter version 7.0.0.3 and WebSphere Adapter for PeopleSoft Enterprise version 7.5.0.3, the following error message is displayed "The version of CWYBS_AdapterFoundation.jar is not compatible with IBM WebSphere Adapter for PeopleSoft Enterprise" as IBM WebSphere Adapter for PeopleSoft Enterprise loads file:/C:/IBM/WebSphere/ ProcServer7/profiles/ProcSrv01/installedConnectors/CWYOE_OracleEBS.rar/ CWYBS_AdapterFoundation.jar with version 7.0.0.3. However, the base level of this jar required is version 7.5.0.3. To overcome this conflict, you must migrate all adapters to the same version level.

There are occasions when you work with embedded adapters that do not need a client-server communication, stand-alone adapters that need a server connection, or a hybrid mix of adapter connections.

The following scenarios cover the different behaviors of AFC version conflict detection, when you are deploying two or more adapters and at least one of the adapter version is 7.5 or higher.

Deploying a stand-alone Adapter
1. Install WebSphere Adapter for Flat Files version 7.0.1.0 through the IBM Business Process Manager administrative console.
2. Install WebSphere Adapter for SAP Software version 7.5.0.0 through the administrative console.
3. Create ActivationSpec for an ALE pass-through inbound operation.
4. Create an application in IBM Integration Designer for a stand-alone ALE pass-through inbound operation.
5. Install and start the application through the administrative console.
6. Verify the error.

Note: An error message is generated in the log/trace area of IBM Business Process Manager, to indicate an AFC version conflict.

Deploying an embedded Adapter
1. Import a build of WebSphere Adapter for FTP version 7.0.1.0, using a RAR file.
2. Create a FTP Inbound EMD operation.
3. Import a build of WebSphere Adapter for Oracle E-Business Suite version 7.5.0.0, using a RAR file.
4. Create an Oracle inbound EMD operation, in the same module where you have created the FTP Inbound EMD operation.
5. Deploy the module to IBM Business Process Manager.
6. Check the trace.

At step 5, the deployment fails. At step 6, you get an internal error message due to the AFC version conflict.

Note: To avoid a name conflict between the business object generated by the two adapters, generate the artifacts into different folders.

Deploying a combination of stand-alone and embedded Adapters
1. Install WebSphere Adapter for JDBC version 7.0.1.0 through theIBM Business Process Manager administrative console.

2. Create an ActivationSpec for a JDBC inbound operation.
3. Create an application in IBM Integration Designer for a JDBC inbound operation, for the stand-alone Adapter deployment.
4. Deploy the JDBC inbound application and trigger your inbound events.
5. Create an application in IBM Integration Designer for a WebSphere Adapter for SAP Software version 7.5.0.0 inbound embedded Adapter deployment.
6. Deploy an SAP inbound application, and trigger your inbound events.

Note: You can resolve the AFC version conflict by using different class loaders for the stand-alone and embedded deployments. With this approach, the migration process handles different CWYBS_AdapterFoundation.jar files and do not conflict with each other. You can start both JDBC and SAP inbound applications successfully, and process Inbound events without exception.

For further assistance, visit http://www.ibm.com/support/docview.wss?uid=swg27006249.

WebSphere Adapters in clustered environments

You can improve adapter performance and availability by deploying a module on a clustered server environment. Clusters are groups of servers that are managed together to balance workloads and to provide high availability and scalability.

The module you deployed is replicated across all servers in a cluster, regardless of whether you deploy the module using a stand-alone or an embedded adapter. The following IBM products support WebSphere Adapters in a clustered environment:
- IBM Business Process Manager or WebSphere Enterprise Service Bus
- WebSphere Application Server Network Deployment
- WebSphere Extended Deployment

To deploy and configure WebSphere Adapter for PeopleSoft Enterprise in a clustered environment, see: "Deploying the module in a clustered environment" on page 72. When you set up a server cluster, you create a Deployment Manager profile. The HAManager, a subcomponent of the Deployment Manager, notifies the Java Platform, Enterprise Edition (JEE) Connector Architecture (JCA) container to activate an adapter instance. For information about creating clustered environments, see the following link: http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=/com.ibm.wbpm.imuc.z.doc/topics/tins_zos_create_cluster.html.

Using WebSphere Extended Deployment, you can optionally enhance the performance of adapter instances in your clustered environment. WebSphere Extended Deployment extends the WebSphere Application Server Network Deployment capabilities by using a dynamic Workload Manager instance instead of a static Workload Manager. The dynamic Workload Manager instance can optimize the performance of adapter instances in the cluster by dynamically balancing the load of the requests. It means that application server instances can be automatically stopped and started based on the load variations, allowing systems with different capacities and configurations to handle load variations evenly. For information about the benefits of WebSphere Extended Deployment, see http://publib.boulder.ibm.com/infocenter/wxdinfo/v6r1m1/index.jsp.

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

High availability for inbound processes

Inbound processes are based on events triggered as a result of updates to data in the PeopleSoft Enterprise server. WebSphere Adapter for PeopleSoft Enterprise is configured to detect updates by polling an event table. The adapter then publishes the event to its endpoint.

When you deploy a module to a cluster, the Java Platform, Enterprise Edition (JEE) Connector Architecture (JCA) container checks the enableHASupport resource adapter property. If the value for the enableHASupport property is true, which is the default setting, all of the adapter instances are registered with the HAManager with a policy 1 of N. This policy means that only one of the adapter instances starts polling for events. 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.

Note: In the active-passive configuration mode of the adapters, the endpoint application of the passive adapter instance also listens to the events/messages even if the `enableHASupport` property is set to `True`. This is because the `alwaysactivateAllMDBs` property in the JMS activation specification is set to `True`. To stop the endpoint application of the passive adapter instance from listening to the events, you must set the `alwaysactivateAllMDBs` property value to `False`. For more information, see "Disabling end point applications of the passive adapter" on page 105.

Important: Do not change the setting of the enableHASupport property.

High availability for outbound processes

In clustered environments, multiple adapter instances are available to perform outbound process requests. So, if your environment has multiple applications that interact with WebSphere Adapter for PeopleSoft Enterprise for outbound requests, then you might improve performance by deploying the module to a clustered environment. In a clustered environment, multiple outbound requests can be processed simultaneously, so that they are not attempting to process the same record.

If multiple outbound requests are attempting to process the same record, such as a Customer address, the workload management capability in WebSphere Application Server Network Deployment distributes the requests among the available adapter instances in the sequence they were received. As a result, these types of outbound requests in a clustered environment are processed in the same manner as in a single server environment: one adapter instance processes only one outbound request at a time. For more information about workload management, see the following link: http://publib.boulder.ibm.com/infocenter/wasinfo/v8r0/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm.html.

Migrating to version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise

By migrating to version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise, you automatically upgrade from the previous version of the adapter. Additionally, you can migrate your applications that embed an earlier version of the adapter, so that the applications can use features and capabilities present in version 7.5.0.3.

Migration considerations

WebSphere Adapter for PeopleSoft Enterprise version 7.5.0.3 may have some features and updates that might affect your existing adapter applications. Before migrating applications that use WebSphere Adapter for PeopleSoft Enterprise, you must consider some factors that might affect your existing applications.

Compatibility with earlier versions

WebSphere Adapter for PeopleSoft Enterprise version 7.5.0.3 is fully compatible with the custom business objects (XSD files) and data bindings that are created using the adapter version 6.1.x, version 6.2.x, version 7.0.x, version 7.5.0, and version 7.5.0.2 and enables the existing business objects and data bindings to work well in the latest version of the adapter.

Because version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise is fully compatible with version 6.1.x, version 6.2.x, version 7.0.x, version 7.5.0, and version 7.5.0.2, any of your applications that used previous versions of WebSphere Adapter for PeopleSoft Enterprise run unchanged when you upgrade to version 7.5.0.3. However, if you want your applications to use features and functionality present in version 7.5.0.3 of the adapter, perform the migration of the artifacts as well as the upgrade of the adapter.

The migration wizard replaces (upgrades) version 6.1.x, version 6.2.x, version 7.0.x, version 7.5.0, or version 7.5.0.2 of the adapter with version 7.5.0.3 and enables version 7.5.0.3 features and functionality for use with your applications.

Note: The migration wizard does not create components or modify existing components, such as mappers and mediators to work with version 7.5.0.3 of the adapters. If any of your applications embed an adapter that is version 7.5.0.2 or earlier and you are upgrading to version 7.5.0.3, and you want your applications to take advantage of the features and functions in version 7.5.0.3, you might need to change those applications.

If the artifacts within a module have inconsistent versions, the entire module is marked as unavailable for migration and cannot be selected. Version inconsistencies are recorded in the workspace log, as they indicate that a project might be corrupted.

The adapter migration wizard in IBM Integration Designer Version 7.5.1 only supports the migration of adapters from version 6.1.x, version 6.2.x, version 7.0.x, version 7.5.0, and version 7.5.0.2 to version 7.5.0.3. It does not support the adapter migration from lower versions to any of the versions before version 7.5.0.3.

Deciding whether to upgrade or to upgrade and migrate

The default processing of the migration wizard is to perform an upgrade of the adapter and to migrate the application artifacts so that the applications can use

features and functions in version 7.5.0.3 of the adapter. When you choose to upgrade the adapter by selecting a project, the wizard automatically selects the associated artifacts for migration.

If you decide that you want to upgrade the adapter from version 6.1.x, version 6.2.x, version 7.0.x, version 7.5.0, or version 7.5.0.2 to version 7.5.0.3, but you do not want to migrate the adapter artifacts, you can do so by clearing the adapter artifacts from the appropriate area of the migration wizard.

Running the migration wizard without selecting any adapter artifacts installs and upgrades your adapter. As the artifacts are not migrated, your applications cannot take advantage of the features and capabilities that exist in version 7.5.0.3 of the adapter.

Migrating multiple adapters referred within a project

When a module contains one or more connector projects, each of which references to different adapters (for example, a module project that contains connector projects referring to JDBC and SAP adapters), the migration wizard identifies the artifacts belonging to each adapter and migrates these artifacts without disrupting the artifacts of other adapters.

When you select the module project and launch the migration wizard:
* The **Source connector** field lists the connector projects with the selected module project.
* The **Dependent artifact projects** area lists only the selected module project.

If you select the connector project and launch the migration wizard:
* The **Source connector** field lists only the selected connector project.
* The **Dependent artifact projects** area lists all projects which reference the selected connector project, including the module project.

Run the migration wizard in a test environment

Because adapter migration might require you to change those applications that use version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise, you must always perform the migration in a development environment first and test your applications before deploying the application to a production environment.

The migration wizard is fully integrated with the development environment.

Performing the migration

You can migrate a project or EAR file to version 7.5.0.3 using the adapter migration wizard. When the tool is finished, the migration is complete and you can work in the project or deploy the module.

Before you begin

Review the information in *Migration considerations*.

About this task

To perform the migration in IBM Integration Designer, complete the following steps.

Note: After migration is complete, the module is no longer compatible with previous versions of the IBM Business Process Manager or WebSphere Enterprise Service Bus runtimes or IBM Integration Designer.

The following steps describe how to run the adapter migration wizard from the connector project menu while in the Java EE perspective in IBM Integration Designer.

Procedure

1. Import the PI (project interchange) file for an existing project into the workspace.

   Note: Ensure that you do not modify the contents of the RAR or copy the adapter JAR file outside the connector project.
2. When projects are created in an earlier version of IBM Integration Designer, the Workspace Migration wizard starts automatically and selects the projects to migrate. Follow the wizard and complete the workspace migration. For more information, see http://publib.boulder.ibm.com/infocenter/dmndhelp/ v7r5m1/topic/com.ibm.wbpm.wid.imuc.doc/topics/tmigsrcart.html.
3. Change to the Java EE perspective.
4. Right-click the module and select **Migrate connector project**. For example, the adapter RAR module.

   You can also launch the adapter migration wizard in the following ways:

   - Right-click the project in the Java EE perspective and select **Migrate adapter artifacts**.
   - From the Problems view, right-click a migration-specific message and select **Quick Fix** to correct the problem.
5. In the Select Projects window, perform the following steps:

   a. The **Source connector** field displays the name of the connector project that you are migrating. If you are migrating a module project, this field lists all the connector projects in the module project. Select the source project from the list. For more information, see "Migrating multiple adapters referred within a project" on page 20.

   b. The **Target connector** field displays the name of the connector to which you are migrating. If you are working with more than one adapter version, this list displays the names of all the compatible connectors. Select the connector you want to migrate.

   c. The **Target version** field displays the version corresponding to the target connector that you selected in the previous step.

   d. The **Dependent artifacts project** area lists the adapter artifacts that are migrated. If you are migrating a module project, this area lists only the selected module project. If you are migrating a connector project within the module project, this area lists all projects which reference the selected connector project, including the module project. By default, all the dependent artifact projects are selected. If you do not select a dependent artifact project, that project is not migrated. You can migrate any project that you have not selected at a later time. Previously migrated projects, projects with a current version, and projects that contain errors are unavailable for migration and are not selected. For more information, see "Upgrading but not migrating a project" on page 22.

   e. Click **Next**. A warning window is displayed with the message, "Properties that are not supported in this version of the target adapter will be removed during the migration".

f. Click **OK**.

6. In the Review Changes window, review the migration changes that occur in each of the artifacts that you are migrating. To view the details, expand each node by clicking the + sign.

7. To complete the migration:

   - Click **Finish**.

   - If the files that need to be updated during migration are in read-only mode, you will be unable to click on the **Finish** button. To view these files, click **Next**. The Update Read-only files window displays the read-only files. To update these files and continue with the migration, click **Finish**. To exit the wizard without migrating the adapter, click **Cancel**.

   Before running the migration process, the wizard performs a backup of all projects affected by the migration. The projects are backed up to a temporary folder within the workspace. If the migration fails for any reason, or if you decide to cancel the migration before it completes, the wizard deletes the modified projects and replaces them with the projects stored in the temporary folder.

   Upon completing the migration successfully, all backed up projects are deleted.

8. If you are migrating an EAR file, optionally create a new EAR file with the migrated adapter and artifacts, and deploy it to IBM Business Process Manager or WebSphere Enterprise Service Bus. For more information about exporting and deploying an EAR file, see the topics devoted to it in this documentation.

Results

The project or EAR file is migrated to version 7.5.0.3. You do not need to run the external service wizard after exiting the adapter migration wizard.

Upgrading but not migrating a project

You can upgrade the adapter from an earlier version, to version 7.5.0.3 while choosing not to migrate the adapter project artifacts.

About this task

Running the migration wizard without selecting any adapter artifacts installs and upgrades your adapter. As the artifacts are not migrated, your applications cannot take advantage of the features and capabilities that exist in version 7.5.0.3 of the adapter.

Procedure

1. Import the PI (project interchange) file into the workspace.

2. When projects are created in an earlier version of IBM Integration Designer, the Workspace Migration wizard starts automatically and selects the projects to migrate. Follow the wizard and complete the workspace migration. For more information, see http://publib.boulder.ibm.com/infocenter/dmndhelp/ v7r5m1/topic/com.ibm.wbpm.wid.imuc.doc/topics/tmigsrcart.html.

3. In the Java EE perspective, right-click the project name and click **Migrate connector project**. The **Adapter Migration** wizard is displayed.

4. In the Select Projects window, clear the dependent artifact projects, and click **Next**. A warning window is displayed with the message, "The properties that are not supported in the version of the target adapter will be removed during the migration."

5. Click **OK**.
6. In the Review Changes window, review the migration changes that occur during updating the project. To view the details, expand each node by clicking the + sign.
7. To complete the migration:
   - Click **Finish**.
   - If the files that need to be updated during migration are in read-only mode, you will be unable to click on the **Finish** button. To view these files, click **Next**. The Update Read-only files window displays the read-only files. To update these files and continue with the migration, click **Finish**. To exit the wizard without migrating the adapter, click **Cancel**.

Results

The project can now be used with WebSphere Adapter for PeopleSoft Enterprise, version 7.5.0.3.

Migrating WebSphere Business Integration applications

You need to migrate the WebSphere Business Integration applications so that they become compatible with Version 7.5.0.3 of your adapter.

About this task

Migrating WebSphere Business Integration applications for use with Version 7.5.0.3 of your WebSphere adapter is a multistep process. First, the artifacts from WebSphere InterChange Server are migrated and converted. A project is then created for the artifacts in IBM Integration Designer. In the remaining steps, the adapter-specific artifacts are migrated and converted into the JCA-compliant format supported by Version 7.5.0.3 of the adapter.

Example

The following diagram shows the wizards that you use to migrate WebSphere Business Integration solutions from WebSphere InterChange Server, so that these applications can be used with Version 7.5.0.3 of your adapter.

Migrating WebSphere Business Integration solutions

IBM Integration Designer

WebSphere InterChange Server Installation directory

1 | WebSphere Interchange Server migration wizard | The imported jar file becomes a project in **IBM Integration Designer - The adapter artifacts are not fully JCA-compliant at this point**

Imports/converts WebSphere InterChange artifacts

Jar file

2 | Enterprise service wizard | Run the External service wizard to import the connector project

3 | Adapter migration wizard | Run the Adapter migration wizard to convert WBI adapter artifacts into JCA-compliant artifacts for use within IBM Business Process Manager

Migrating applications from WebSphere InterChange Server

To use Version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise with applications from WebSphere InterChange Server, you need to migrate the application artifacts and convert them so that they can be deployed and run on IBM Business Process Manager or WebSphere Enterprise Service Bus. Understanding this task at a high level helps you perform the steps that are needed to accomplish the task.

The following figure illustrates the flow of the migration task. The steps that follow the figure describe this task at a high level only. See the topics following this roadmap for the details on how to perform each of these steps.
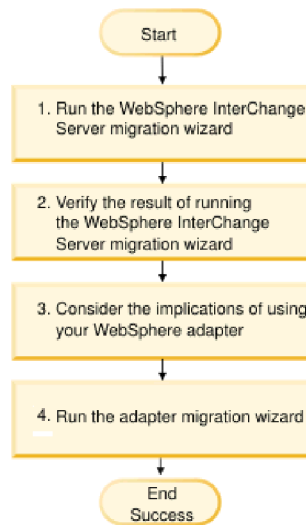
Figure 2. Roadmap for migrating applications from WebSphere InterChange Server

Migrating applications from WebSphere InterChange Server

This task consists of the following steps:

1. Run the WebSphere InterChange Server migration wizard.

   The WebSphere InterChange Server migration wizard moves the application artifacts into IBM Integration Designer. The migrated adapter artifacts are not fully JCA-compliant at the completion of this task.

2. Verify that the WebSphere InterChange Server migration is successful.

   Review all messages from the Migration results window and take action if required.

3. Consider the implications of using Version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise.

   In addition to considerations for migrating WebSphere InterChange Server applications, you need to consider how Version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise works with the migrated applications. Some of the adapter operations supported by WebSphere InterChange Server applications might be supported and implemented differently with Version 7.5.0.3 of the adapter.

4. Run the adapter migration wizard.

   Run the adapter migration wizard to update adapter-specific artifacts such as the schemas and service definition files (.import, .export, and .wsdl files) for use with Version 7.5.0.3 of the adapter.

Migration considerations for WebSphere Business Integration adapters

By migrating to WebSphere Adapter for PeopleSoft Enterprise Version 7.5.0.3, you have an adapter that is compliant with the Java Platform, Enterprise Edition (JEE) Connector Architecture (JCA) and designed specifically for service-oriented architecture.

Note: WebSphere Business Integration adapter and JCA-compliant adapter use different event Component Interface. You must change to the new event Component Interface supported by the JCA-compliant adapter manually after the completion of migration.

Operations considerations

Several WebSphere Business Integration adapter operations are supported differently in WebSphere Adapter for PeopleSoft Enterprise Version 7.5.0.3. You might need to provide additional development effort when you plan to use the following operations:

DeltaUpdate
> The DeltaUpdate is not supported in WebSphere Adapter for PeopleSoft Enterprise. You must use the ApplyChanges operation to implement delta processing of components.

Exists The Exists operation is implemented differently in WebSphere Adapter for PeopleSoft Enterprise.

RetrieveByContent
> The RetrieveAll operation is supported in WebSphere Adapter for PeopleSoft Enterprise as an equivalent to the RetrieveByContent operation.

Application artifacts

Before running the adapter migration wizard, use the WebSphere InterChange Server migration wizard to generate the application artifacts for the WebSphere Business Integration adapter, including the business objects, maps, and collaborations. Then you can run the adapter migration wizard to update the adapter-specific artifacts such as the schemas and service definition files (.import, .export, and .wsdl) so that they are suitably converted into a format that is compliant with JCA.

Run the migration wizard in a test environment first

Because migrating from a WebSphere Business Integration adapter to WebSphere Adapter for PeopleSoft Enterprise might require changes to the applications that use Version 7.5.0.3 of WebSphere Adapter for PeopleSoft Enterprise, always perform the migration in a development environment first and test your applications before deploying the application to a production environment.

Migrating application artifacts from WebSphere InterChange Server

To migrate the application artifacts into IBM Integration Designer, run the WebSphere InterChange Server migration wizard. The wizard imports and converts most of the artifacts into a format that is compatible with IBM Business Process Manager or WebSphere Enterprise Service Bus.

Before you begin

Launch the WebSphere InterChange Server migration wizard from within IBM Integration Designer to migrate the application artifacts from WebSphere InterChange Server format into artifacts that are compatible with IBM Business Process Manager or WebSphere Enterprise Service Bus.

For information about how to prepare to migrate artifacts from WebSphere InterChange Server and for detailed instructions on performing the migration and verifying that the migration was successful, see http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5m1/index.jsp?topic=/com.ibm.wbpm.wid.imuc.doc/topics/twics.html.

About this task

Running WebSphere InterChange Server migration wizard might not fully convert adapter-specific artifacts (such as service descriptors, service definitions, and business objects) into IBM Business Process Manager or WebSphere Enterprise Service Bus compatible artifacts. To complete the migration of adapter-specific artifacts, run the adapter migration wizard after you have successfully run the WebSphere InterChange Server migration wizard.

Note: While you run the WebSphere InterChange Server migration wizard, ensure that you set each connector in the repository to the same adapter version.

Results

The project and application artifacts are migrated and converted into IBM Business Process Manager compatible artifacts.

What to do next

Run the adapter migration wizard to migrate the adapter-specific artifacts.

Migrating adapter-specific artifacts

After a project is created for the artifacts in IBM Integration Designer, you can migrate the project using the adapter migration wizard. The adapter migration wizard updates adapter-specific artifacts such as the schemas and service definition files (.import, .export, and .wsdl) for use with version 7.5.0.3 of the adapter. When you finish running the adapter migration wizard, the migration is complete and you can work in the project or deploy the module.

Before you begin

Before running the adapter migration wizard, do the following steps:
- Review the information in "Migration considerations" on page 19.
- Run the WebSphere InterChange Server migration wizard to migrate the project and convert data objects for use with IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

After migration is complete, the module will work only with Version 7.5.0.3 of your adapter.

To perform the migration in IBM Integration Designer, complete the following steps.

Procedure

1. Import the PI (project interchange) file for an existing project into the workspace.

2. When projects are created in an earlier version of IBM Integration Designer, the Workspace Migration wizard starts automatically and selects the projects to migrate. Follow the wizard and complete the workspace migration. For more information, see http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5m1/topic/com.ibm.wbpm.wid.imuc.doc/topics/tmigsrcart.html.

3. Change to the Java EE perspective.

4. Right-click the connector project and select **Migrate connector project**.

   You can also launch the adapter migration wizard by using the right-click option and selecting the module project in the Java EE perspective and selecting **Migrate adapter artifacts**.

   Note:

   If the adapter type (for example, CICS/IMS adapter) is not supported by the migration wizard, the **Migrate connector project** and **Migrate adapter artifacts** menus are not available for selection. If the adapter project is of the latest version and the module projects referencing this adapter project are also of the latest version, these menus are disabled.
   When you launch the migration wizard from the connector project while in the Java EE perspective, by default all the dependent artifact projects are selected. If you do not select a dependent artifact project, that project is not migrated.

5. In the Select Projects window, perform the following steps:
   a. The **Source connector** field displays the name of the connector project that you are migrating. Select the source project from the list.
   b. The **Target connector** field displays the name of the connector to which you are migrating. If you are working with more than one adapter version, this list displays the names of all the compatible connectors. Select the connector to which you want to migrate.
   c. The **Target version** field displays the version corresponding to the target connector you selected in the previous step.
   d. The **Dependent artifacts project** area lists the adapter artifacts that are migrated.
   e. Review the tasks and warnings presented on the welcome page, and click **Next**. A warning window is displayed with the message, "The properties that are not supported in the version of the target adapter are removed during the migration."
   f. Click **OK**.

6. In the Review Changes window, review the migration changes that occur in each of the artifacts that you are migrating. To view the details, expand each node by clicking the + sign.

7. To complete the migration:
   - Click **Finish**.
   - If the files that need to be updated during migration are in read-only mode, you will be unable to click on the **Finish** button. To view these files, click **Next**. The Update Read-only files window displays the read-only files. To

update these files and continue with the migration, click **Finish**. To exit the wizard without migrating the adapter, click **Cancel**.

Before performing the migration process, the wizard backs up all projects affected by the migration. The projects are backed up to a temporary folder within the workspace. If the migration fails for any reason, or if you decide to cancel the migration before it completes, the wizard deletes the modified projects and replaces them with the projects stored in the temporary folder.

8. Select **Project > Clean**, to refresh and rebuild the workspace for the changes to take effect.

9. On successful migration, all backed up projects are deleted. Remove the Sync inbound flow manually as this flow is not used by the adapter. From the migrated project, select the Input_Sync inbound flow, right-click, and select **Delete**.
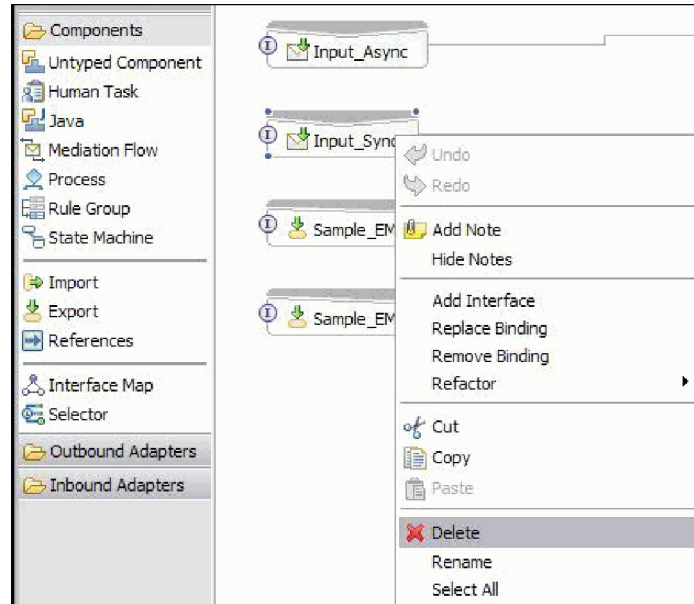
10. If you are migrating an EAR file, create a new EAR file with the migrated adapter and artifacts, and deploy it to IBM Business Process Manager or WebSphere Enterprise Service Bus. For information about exporting and deploying an EAR file, see "Deploying the module for production" on page 68.

Results

The project is migrated to Version 7.5.0.3. You do not need to run the external service wizard after exiting the adapter migration wizard.

Changes to the import, export, and WSDL files after migration

When the WebSphere InterChange Server migration wizard moves the application artifacts into IBM Integration Designer, changes made are reflected in the service definition files: the import, export, and WSDL files.

The migrated adapter artifacts are not fully JCA-compliant at the completion of this task. You can complete the migration of the adapter-specific artifacts (such as service descriptors, service definitions, and business objects) to a JCA compatible format by running the adapter migration wizard.

Changes to the import file

During migration, the affected module artifacts are migrated to an import file. The existing JMS Binding property is changed to the EIS Binding property in the import file. The other property details added in the import file include information about the data binding configuration, changes to the connection information in the Managed Connection Factory properties, and several new method bindings.

Changes to the export file

During migration, the affected module artifacts are migrated to an export file. The existing JMS Binding property is changed to the EIS Binding property in the export file. The other property details added in the export file include information about the data binding configuration, changes to the connection information in the

Activation Specification properties, and several new method bindings.

Changes to the WSDL file after migration

During migration, the affected module artifacts are migrated to corresponding WSDL files that include PeopleSoft specific service description WSDL artifacts. The service description files become JCA compatible. The WSDL files have an input and output type for each operation. Both the inbound and outbound operations work on their specific input types to produce corresponding output types after the operations are performed.

Note:

- When you migrate multiple top level inbound business objects in the project, only the first top-level business object inbound feature works correctly. For the other top level inbound business object to work correctly, you must manually modify the "emit + [verb name] + after image + [business object name]" method in the Input_Processing.java and Input_Async_Processing.java class to call the correct destination services.
- The WebSphere Business Integration Adapter for PeopleSoft properties that are either not valid or not supported by WebSphere Adapter for PeopleSoft Enterprise are removed from the migrated artifacts.

Chapter 3. Samples and tutorials

To help you use WebSphere Adapters, samples and tutorials are available from the Business Process Management Samples and Tutorials website.

You can access the samples and tutorials in either of the following ways:

- From the welcome page of IBM Integration Designer, click **Go to Samples and Tutorials**. In the Samples and Tutorials pane, under More samples, click **Retrieve**. Browse the displayed categories to make your selection.
- From the Business Process Management Samples and Tutorials website: http://bpmwikilogin.blueworkslive.com/accounts/login/.

Chapter 4. Configuring the module for deployment

To configure the adapter so that it can be deployed on IBM Business Process Manager or WebSphere Enterprise Service Bus, use IBM Integration Designer to create a module, which is exported as an EAR file when you deploy the adapter. You then specify the business objects you want to discover and the system on which you want to discover them.

Roadmap for configuring the module

Before you can use WebSphere Adapter for PeopleSoft Enterprise in a runtime environment, you must configure the module. Understanding this task at a high level helps you perform the steps that are needed to accomplish the task.

You configure the module for the adapter to use by using IBM Integration Designer. The following figure illustrates the flow of the configuration task, and the steps that follow the figure describe this task at a high level only. See the topics that follow this road map for the details on how to perform each of these steps.
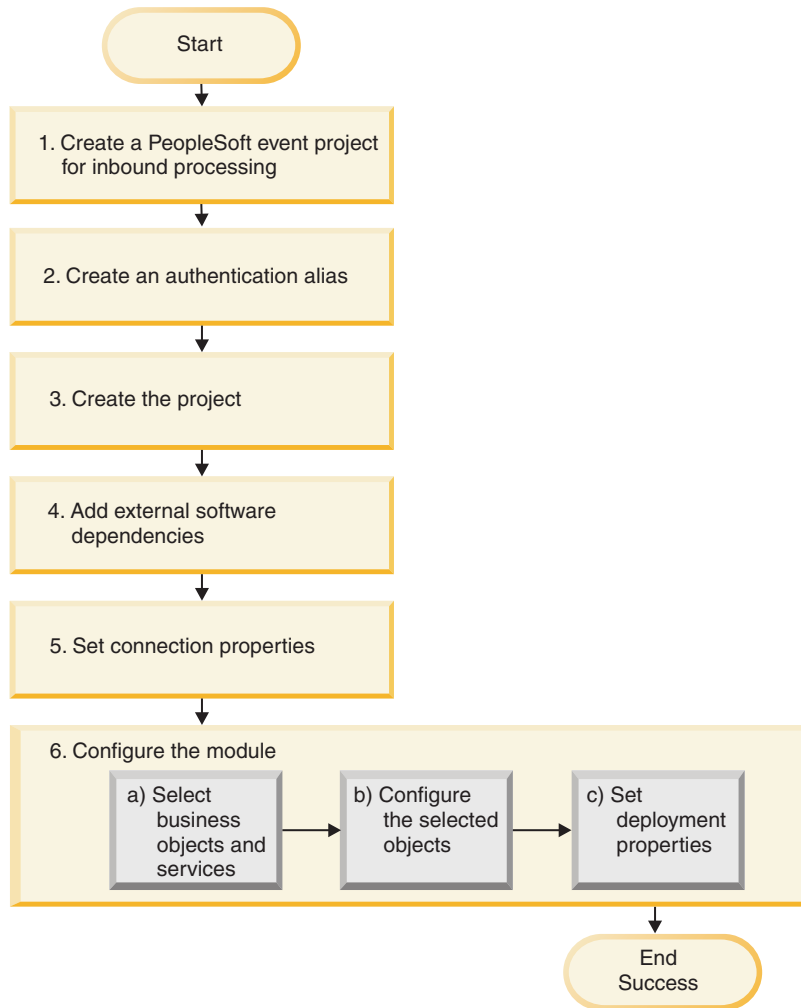
Figure 3. Roadmap for configuring the module

Configuring the module for deployment environment

This task consists of the following step:

1. Create a PeopleSoft event project for inbound processing.
2. Create an authentication alias to access the PeopleSoft Enterprise server with an encrypted password. This step is optional, depending on your policy for handling passwords and IDs. You perform this step using the server.
3. Create the project. First, start the external service wizard in IBM Integration Designer to begin the process of creating and deploying a module. The wizard creates a project that is used to organize the files associated with the module.
4. Add the external software dependencies required by WebSphere Adapter for PeopleSoft Enterprise to the project. These dependencies are also required when you export the module as an EAR file, and deploy the EAR file to the server.
5. Set connection properties that the external service wizard needs to connect to the PeopleSoft Enterprise server for discovery of objects and services.
6. Configure the module for inbound processing or outbound processing by using the external service wizard to find and select business objects and services from the PeopleSoft Enterprise server, and to generate business object definitions and related artifacts.

a. Select business objects and services for inbound or outbound processing from the business integration components discovered by the external service wizard.

b. Configure the selected objects by specifying operations and other properties that apply to all the business objects.

c. Set deployment properties that the adapter uses to connect to the PeopleSoft Enterprise server at run time. Then, generate the service by using the external service wizard to save the new module, which contains the business object or objects you configured, the import file or export file, and the service interface.

Creating a custom event project in PeopleTools

To perform asynchronous inbound event processing, the adapter requires an event project in PeopleSoft. Use PeopleTools to create the custom event project.

About this task

If your environment requires inbound event support, you must use a custom event project in PeopleSoft. A sample event project, IBM_EVENT_V600, is provided with the adapter. You can either modify and use the sample project, or you can create your own project using PeopleTools. If you create your own project, use the following steps:

Procedure

1. Use PeopleTools Application Designer to create and name a new project.
2. Create the fields for the new project as described in the following table:

Field name	Field description
IBM_EVENT_ID	Numeric value that is retrieved from the IBM_FETCH_ID record. This value is a unique ID for the event.
IBM_OBJECT_NAME	Name of the corresponding business graph.
IBM_OBJECT_KEYS	The get key property names in the Component Interface followed by the key values in name-value pairs. This value is used to retrieve the component from the enterprise information system (EIS).
IBM_EVENT_STATUS	Status of the event. The status is 0 if the event is ready to be polled. The IBMPublishEvent function is called.
IBM_OBJECT_VERB	Verb that is set on the business object graph that contains the retrieved business object.
IBM_EVENT_DTTM	Date on which the event is created. For a future dated event, this value is the effective date.
IBM_NEXT_EVENT_ID	Field that has the latest event ID under the record IBM_FETCH_ ID. The value is incremented for each event that is added to the IBM_EVENT_TBL table. The IBM_EVENT_ID field is populated within that table.

Field name	Field description
IBM_XID	The transaction ID which ensures providing of assured event delivery.
IBM_CONNECTOR_ID	The unique identifier for the adapter instance that receives a specific event.

3. Create a record named IBM_EVENT_TBL and add to it all fields you created except IBM_NEXT_EVENT_ID.
4. Create a record named IBM_FETCH_ID and add to it only the IBM_NEXT_EVENT_ID field.
5. Open the IBM_FETCH_ID record, select the IBM_NEXT_EVENT_ID field, view the PeopleCode, and select **fieldformula.**
6. Copy the PeopleCode for custom event project, found under the Reference section of this documentation, to the project you are creating.
7. Create a page under your project that contains the fields of the IBM_EVENT_TBL record at level 0. The page can have any name.
8. Create a component under your project that contains the page you created. The component can have any name.
9. Create a Component Interface against this component and give it any name. Make sure that you use the properties based on the underlying component definition.
10. Build the entire project, selecting all create options.
11. Test and confirm that the Component Interface works, using the Component Interface tester.
12. Generate the Java APIs for the Component Interface, and then add the generated classes to the adapter class path. For more information about building a PeopleTools project and testing the PeopleSoft Component Interface, refer to the PeopleSoft documentation.

Creating an authentication alias

An authentication alias is a feature that encrypts the password used by the adapter to access the PeopleSoft Enterprise server. The adapter can use it to connect to the PeopleSoft Enterprise server instead of using a user ID and password stored in an adapter property.

Before you begin

To create an authentication alias, you must have access to the administrative console. The following procedure shows you how to gain access to the administrative console through IBM Integration Designer. If you are using the administrative console directly (without going through IBM Integration Designer), log in to the administrative console and skip to step 2 on page 37.

About this task

An authentication alias is a feature that encrypts the password used by the adapter to access the PeopleSoft Enterprise server. Using an authentication alias eliminates the need to store the password in clear text in an adapter configuration property, where it might be visible to others.

To create an authentication alias, use the following procedure.

Procedure

1. Start the administrative console.

   To start the administrative console from IBM Integration Designer, perform the following steps:

   a. In the Business Integration perspective of IBM Integration Designer, click the **Servers** tab.

   b. If the server does not show a status of **Started**, right-click the name of the server and click **Start**.

   c. Right-click the name of the server, click **Administration**, and click **Run administrative console**.

   d. Log on to the administrative console. If your administrative console requires a user ID and password, type the ID and password and click **Log in**. If the user ID and password are not required, click **Log in**.

2. In the administrative console, click **Security** > **Secure administration, applications, and infrastructure**.

3. Under **Authentication**, expand **Java Authentication and Authorization Service** and click **J2C authentication data**

4. Create an authentication alias:

   a. In the list of J2C authentication aliases that is displayed, click **New**.

   b. In the **Configuration** tab, type the name of the authentication alias in the **Alias** field.

   c. Type the user ID and password that are required to establish a connection to the PeopleSoft Enterprise server.

   d. Optionally type a description of the alias.

   e. Click **OK**.

      The newly created alias is displayed.

      The full name of the alias contains the node name and the authentication alias name you specified. For example, if you create an alias on the node widNode with the name ProductionServerAlias, then the full name is widNode/ProductionServerAlias. This full name is the one you use in subsequent configuration windows.

   f. Click **Save**, and then click **Save** again.

Results

You created an authentication alias, which you can use when you configure the adapter properties.

Creating the project

To begin the process of creating and deploying a module, you start the external service wizard in IBM Integration Designer. You then create a project, which is used to organize the files associated with the adapter.

Before you begin

Make sure that you have gathered the information you establish a connection to the PeopleSoft Enterprise server. For example, you need the name (or IP address) of the PeopleSoft Enterprise server and the user ID and password to access the PeopleSoft Enterprise server.

About this task

Start the external service wizard and create a project for the adapter in IBM
Integration Designer. If you have an existing project, you can select it instead of
creating one.

To start the external service wizard and create a project, use the following
procedure.

Procedure
1. To start the external service wizard, go to the Business Integration perspective
   of IBM Integration Designer, and then click **File** > **New** > **External Service**.
2. In the Select the Service Type or Registry window, expand the **Adapters** node
   and select **PeopleSoft**.
3. Click **Next**.
4. From the Select an Adapter window, create a project or select an existing
   project.
   - To create a project, perform the following steps:
     a. Select **IBM WebSphere Adapter for PeopleSoft Enterprise (IBM :
        version)**, where *version* is the version of the adapter you want to use and
        click **Next**.
     b. In the Import a RAR File window, accept the default project name in
        Connector project or type a different name.
   - To select an existing project, perform the following steps:
     a. Expand **IBM WebSphere Adapter for PeopleSoft Enterprise (IBM :
        version)**.
     b. Select a project.

        For example, if you have an existing project named
        CWYES_PeopleSoftAdapter, you can expand **IBM WebSphere Adapter
        for PeopleSoft Enterprise (IBM : version)** and select the project.
     c. Click **Finish**.

Results

A new project is created and is listed in the Business Integration window.

What to do next

Provide the location of the psjoa.jar file and other required files.

Adding external software dependencies

The external service wizard needs two files either copied or generated from the
PeopleSoft Enterprise to be able to communicate with it. Use the external service
wizard to specify the location of the JAR files. To add the required JAR files to
your project, make sure that the files are in the local machine and then use the
IBM Integration Designer to specify the location of the files.

Before you begin

You need to be running the external service wizard in IBM Integration Designer to
perform this task.

About this task

To add the two required PeopleSoft dependency files, the PeopleSoft psjoa.jar and the component interface JAR file, use the Locate the Required Files and Libraries window of the external service wizard to specify the location of the files.

Procedure

1. Obtain the two PeopleSoft driver-specific files from your PeopleSoft Enterprise administrator.
2. In the Locate the Required Files and Libraries window, specify the location of the PeopleSoft driver-specific files.
3. Click **Next**.

Results

The wizard has the files it needs to communicate with the PeopleSoft server.

Setting connection properties for the external service wizard

To set connection properties for the external service wizard so that it can access the PeopleSoft Enterprise server, specify information such as the user name and password you use to access the server and the name or IP address of the server.

Before you begin

Make sure you have successfully added the software dependency files (the psjoa.jar and associated files).

About this task

Specify the connection properties that the external service wizard needs to establish a connection to the PeopleSoft Enterprise server and discover functions or data.

To specify the connection properties, use the following procedure.

Procedure

1. In the Select the Processing Direction window, select **Inbound** (if you are going to send data from the PeopleSoft Enterprise server) or **Outbound** (if you are going to send data to the PeopleSoft Enterprise server) and click **Next**. The Specify the Discovery Properties window is displayed.
2. In the Specify the Discovery Properties window, specify the configuration properties:
   a. In the **Host name** field, type the name (or IP address) of your PeopleSoft Enterprise server.
   b. In the **Port number** field, type the port number the adapter uses to access the PeopleSoft Enterprise server
   c. Type the name and password you use to access the PeopleSoft Enterprise server.

      The password is case-sensitive.
   d. Optional: Type a prefix to all objects associated with this module.

      A prefix allows you to differentiate between objects generated against the same PeopleSoft component interface.

e. Optional: Select the **Prompt for additional configuration settings when adding business object** check box if you want to configure additional metadata when you generate objects. Use this option if you plan to use Delete and RetrieveAll operations and have the effective-dated fields and generated keys used in the operation.

3. Optional: To set additional advanced properties, click **Advanced**.

4. Optional: To enable bidirectional support for the adapter at run time:

   a. In the **Bidi properties** area, select **Bidi transformation**.

   b. Set the ordering schema, text direction, symmetric swapping, character shaping, and numeric shaping properties to control how bidirectional transformation is performed.

5. Optional: To change the location of the log files for the wizard or the amount of information included in the logs, click **Change the logging properties for the wizard**, and then provide the following information:

   a. In **Log file output location**, specify the location of the log file for the wizard.

   b. In **Logging level**, specify the severity of errors that you want logged.

      Note: This log information is for the wizard only; at run time, the adapter writes messages and trace information into the standard log and trace files for the server.

6. Click **Next**.

Results

The external service wizard contacts the PeopleSoft Enterprise server, using the information you provided (such as user name and password) to log in. The Find Objects in the Enterprise System window is displayed.

What to do next

Specify search criteria that the external service wizard uses to discover functions or data on the PeopleSoft Enterprise server.

Configuring the module for outbound processing

To configure a module to use the adapter for outbound processing, use the external service wizard in IBM Integration Designer to find and select business objects and services from the PeopleSoft Enterprise server, and to generate the business object definitions and related artifacts.

Selecting business objects and services

To specify the objects you want to import, you use the external service wizard.

Before you begin

Make sure that you have set the connection properties for the external service wizard.

About this task

To find and select one or more objects, use the following procedure.

Procedure

1. In the Find Objects in the Enterprise System window, click the object you want to work with, and then click the arrow to move it to the selected set.

   To see children of an object, click the plus sign.

2. In the Specify the Configuration Properties for 'object' window, specify configuration properties for the object. (This window is available only if you select the **Prompt for additional configuration settings when adding a business object** check box on the Specify the Discovery Properties window.)

   a. Select the field name to use for performing logical deletes.

   b. In the **Value used to indicate a deleted object** field, type the status value used to indicate that an object is no longer active.

   c. Optional: If you want to use alternate keys during RetrieveAll operations, click **Add** and select the key fields you want to use.

3. Optional: To set additional parameters, click **Advanced**.

   a. Click the **Additional properties** section, and select from the following options:

      - At the **Preserve objects in parent update operation** box, click **Add** and select the child objects.

      - At the **Field that represents generated keys in PeopleSoft** box, click **Add** and select the fields.

   b. Click the **Effective date configuration** section, and select from the following effective date options:

      - **Retrieve only current effective dated item**: The corresponding ASI is GetCurrentItem.

      - **Insert at oldest effective date position**: The corresponding ASI is InsAtOldestEffDtPos.

      - **Insert at current effective date position**: The corresponding ASI is InsAtCurrentEffDtPos.

      - **Ignore effective sequence number**: The corresponding ASI is IgnoreEffectiveSequence.

      - **Field that represents effective date value**: Enables effectiveDate flag to true or false for EFFDT (effective date) field as ASI.

      - **Field that represents effective date sequence**: Enables effectiveSequence flag to true or false for EFFSEQ (effective sequence) field as ASI.

      For more information about the effective date options, see "Application-specific information" on page 111 (ASI).

4. Click **OK** to add the object to the list of objects to be imported.

5. When you select all objects, click **Next**.

Results

You selected one or more objects to work with.

What to do next

From the Specify Composite Properties window, add or remove associated operations.

Configuring the selected objects

To configure the business object, you specify information about the object (such as the operation associated with the object).

Before you begin

Make sure that you have selected and imported the object.

About this task

To configure the business object, use the following procedure.

Procedure

1. In the Specify Composite Properties window, enter the following information.
   a. Add or remove operations for the object that is added to the service interface.
   b. In the **Maximum number of records for RetrieveAll operations** , specify the maximum number of records to retrieve when processing. The default value is 100.
   c. If you do not want to specify a namespace value, leave the default value for the **Business object namespace field.**
   d. Optional: To indicate where the generated business objects be stored, type the relative path to the folder in the **Folder** field.

      Note: If you are creating multiple adapter artifacts within a module, ensure that you specify different business object folders for each adapter within the module. For example, if you are creating artifacts for Oracle, JDBC, SAP, and JDE within a module, you need to create different relative folders for each of these adapters. If you do not specify different relative folders, the existing artifacts are overwritten when you generate new artifacts.
   e. If you want the business object contained within a business graph, leave the check box selected.
2. Click **Next**.

Results

The Specify the Service Generation and Deployment Properties window is displayed.

What to do next

Generate a module that includes the adapter and the business objects.

Setting deployment properties and generating the service

After you select and configure business objects for your module, you can use the external service wizard to configure properties that the adapter uses to connect to a specific enterprise information system. The wizard creates a business integration module where all the artifacts and property values are saved.

Before you begin

Ensure that you have configured the business object.

About this task

Use the Specify the Service Generation and Deployment Properties and Specify the Location Properties windows of the external service wizard to perform this task. The connection properties in this task are initialized to the values that the wizard used to connect to the enterprise information system. To configure the module to use other values, change the values here.

Procedure

1. In the Specify the Service Generation and Deployment Properties window, click **Edit Operations** if you want to review or modify the names, or add a description for the operations to the business objects that you are creating.

Figure 4. Specify the Service Generation and Deployment Configuration window

a. In the Edit Operations Name window, select the operation you want to edit, and then click **Edit**.

b. In the Add/Edit properties window, type a new operation name and description, and then click **Finish**.

2. Specify how you want the adapter to get the user name and password at run time.

- To use a J2C authentication alias, select **Using an existing JAAS alias (recommended)** and type the name of the alias in **J2C Authentication Data Entry**. You can specify an existing authentication alias or create one at any time before deploying the module. The name is case-sensitive and includes the node name.

- To use managed connection properties, select **Using security properties from managed connection factory** and type the values in the **User name** and **Password** fields.

- If you want to administer the user name and password from other mechanism, select **Other**.

3. In the **Deploy connector project** field, specify whether to include the adapter files in the module. Select one of the following values:

- **With module for use by single application**. When the adapter files are embedded in the module, you can deploy the module to any application server. Use an embedded adapter when you have a single module using the adapter or if multiple modules must run different versions of the adapter. You can use an embedded adapter to upgrade the adapter in a single module without the risk of destabilizing other modules by changing their adapter version.

- **On server for use by multiple applications**. If you do not include the adapter files in a module, you must install them as a stand-alone adapter on each application server where you want to run the module. Use a stand-alone adapter when multiple modules can use the same version of the adapter and you want to administer the adapter in a central location. A stand-alone adapter can also reduce the resources required by running a single adapter instance for multiple modules.

4. If you selected **On server for use by multiple applications** in the previous step, indicate how you want to specify the connection properties.

- If you manually created and configured a managed connection factory or activation specification on the server or if you have already deployed an application that connects to the same enterprise information system using the same managed connection factory or activation specification properties, you can reuse the managed connection factory or activation specification by specifying the name of its Java Naming and Directory Interface (JNDI) data source. The following figure shows the **JNDI Lookup Name** field.

  a. In **Connection settings**, select **Use predefined connection properties**.

  b. In **JNDI Lookup Name**, type the name of the JNDI data source for an existing managed connection factory or activation specification.

  c. Click **Next** to complete this task.

- If this application is the first that connects to the enterprise information system with a specific user name and password, or if you want to administer the user name and password separately from other applications, select **Specify connection properties**.

5. Review and, if necessary, change the values of the required connection properties. The **Host name**, **Port number**, and **Language code** fields are initialized with the connection information that you specified when you

started the wizard. You can also connect to an alternate enterprise information system, although the schema names must be the same in both. The format of the connection properties is specific to the enterprise information system. For more information about the properties, see "Managed connection factory properties" on page 127.

6. Enter the name of the **Component interface for testing failed connection**.

7. After clicking **Advanced**, specify the advanced options.
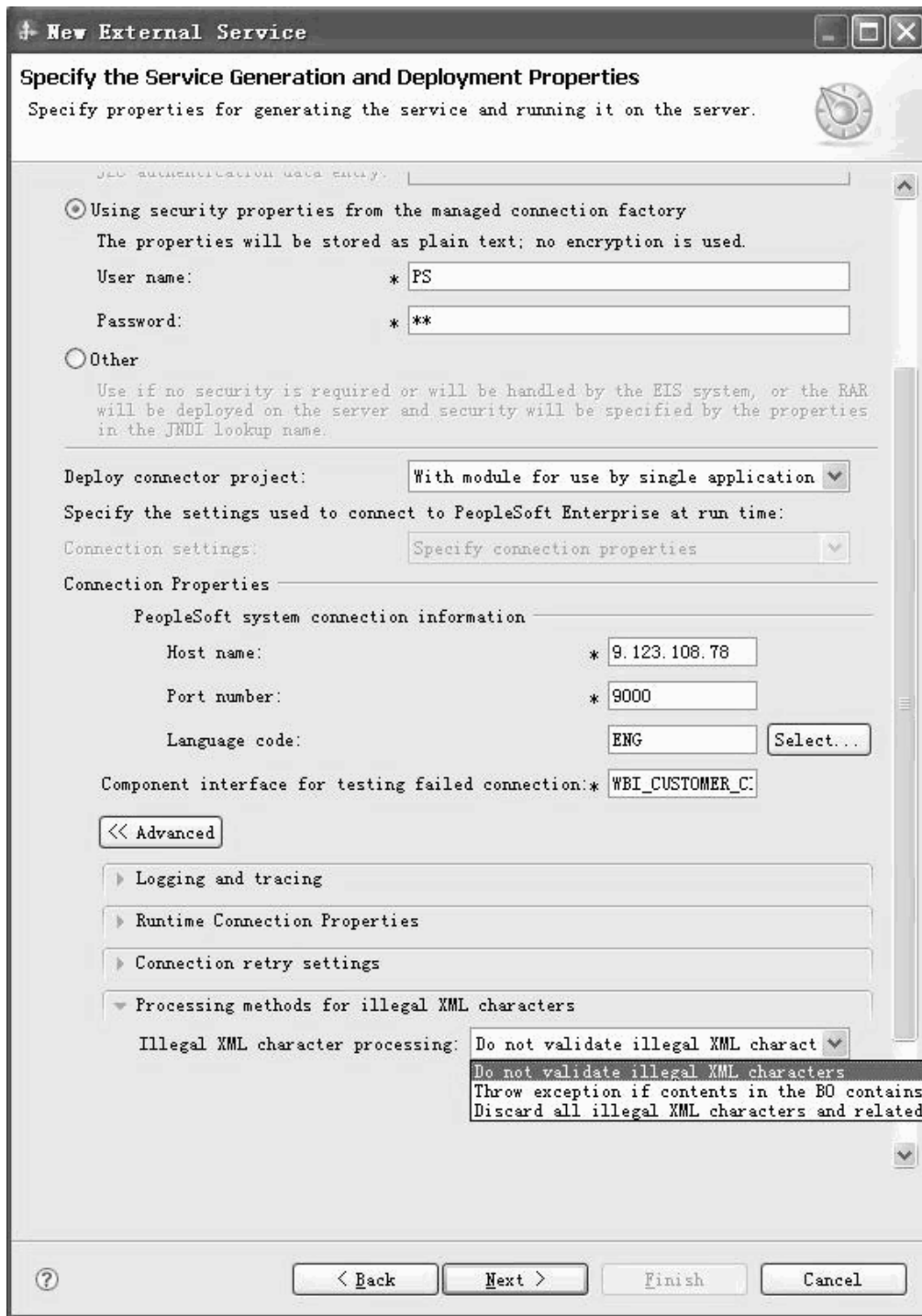
Figure 5. Specify the Service Generation and Deployment Configuration window-Advanced

- **Processing methods for illegal XML characters**
  - a. From the **Illegal XML character processing** drop-down list, select one of the options that correspond to your requirement:
    - **Do not validate illegal XML characters**, if you want to work with the default adapter behavior.

– **Throw exception if contents in the BO contains illegal XML characters**, if you want to proceed after you get an exception message, and log the illegal XML characters in the trace file, at runtime.

– **Discard all illegal XML characters and related logs**, if you want the adapter to discard the illegal XML characters, and log the characters in the trace file, at runtime.
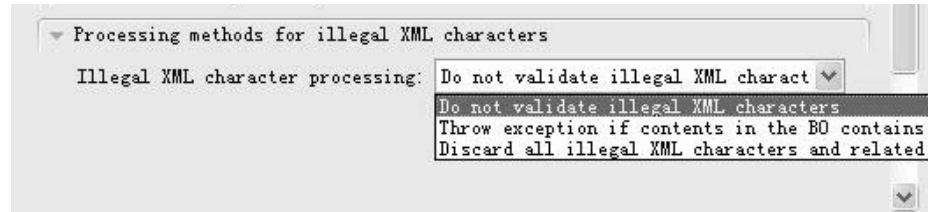
- **Logging and tracing** area:
  a. If you have multiple instances of the adapter, set **Adapter ID** to a value that is unique for this instance. For more information, see "Adapter ID (adapterID)" on page 128.
  b. You can select **Disguise user data as 'XXX' in log and trace files** if you want to prevent sensitive user data from being written to log and trace files. For more information, see "Disguise user data as "XXX" in log and trace files (hideConfidentialTrace) " on page 129.
- **Run time Connection properties** area:
  a. Select **Use Invalid Request Fault**, if you want the adapter to return an invalid request fault business object if any error occurs during run time. For more information, see "Use Invalid Request Fault (useInvalidRequestFault)" on page 132.
  b. Select **Show error on empty result set**, if you want the adapter to generate a RecordNotFoundFault fault when no records are found. For more information, see "Show error on empty result set (showErrorOnEmptyResultSet)" on page 130.
- **Connection retry settings** area:
  a. To specify the number of connection exceptions in the outbound operation, set the **Maximum retries in case of system connection failure:** to a value greater than or equal to 0. For more information, see "Maximum retries in case of system connection failure (connectionRetryLimit)" on page 130.
  b. To specify the time interval between retries if a connection fails, set the **Retry interval if connection drops (in milliseconds):** to a value in milliseconds. For more information, see "Retry interval if connection drops(in milliseconds) (connectionRetryInterval)" on page 132.

8. Click **Next**. The Specify the Location Properties window is displayed.
9. In the Specify the Location Properties window, specify the name of the module that you want to create. This module can be new or existing.
   - If the module name is displayed in the **Module** list, select its name.

     Important: If the module contains an interface or business object with the same name as any that you are now configuring, the original interface or business object in the module is replaced by the new version.
   - Otherwise, in the Specify the Location Properties, create a module:
     a. Click **New**.

b. In the Select a Business Integration Project Type window, select **Create a module project** and click **Next**.

c. In the Create a Module window, type a name for the module. For example, `PeopleSoftOutboundInterface`.

d. If you want the service description files (the `.import` and `.wsdl` files) to be in the default folder in the module, leave **Use default location** selected. If you want to specify a different folder in the module, clear the option and then click **Browse** to specify a different folder in **Location**.

e. If you want the module to be opened automatically in the assembly diagram in IBM Integration Designer when the wizard closes, select **Open module assembly diagram**. Otherwise, clear this option.

f. Click **Finish** to create the module.

10. Specify the namespace that you want to use for your business objects.
    - If you want the business objects in the module to use the default derived namespace, leave **Use default namespace** selected.
    - To specify a different namespace, clear the option and type a different value in **Namespace**.

11. Optional: Specify the folder within the new module where the service description is saved. In **Folder**, type the folder name or browse to an existing folder. If you do not specify a folder name, the artifacts (the import, XSD, and WSDL files) are stored in the root folder of the module, that is, the folder with the module name.

12. In **Name**, accept the default import name or type a different name.

13. Optional: If you want to save the business objects in a library where they can be used by other modules, select **Save business objects to a library** and specify the location of the library in **Library**.

14. Optional: In **Description**, type a descriptive comment about the module.

15. When you are finished setting properties, click **Finish**.

16. If the Model Changed window is displayed, click **Yes**.

Results

The wizard exits. The module is created in the project and artifacts are generated.

What to do next

You might need to use the assembly editor to complete the configuration. Later, you can test or deploy your module.

Configuring the module for inbound processing

To configure a module to use the adapter for inbound processing, use the external service wizard in IBM Integration Designer to find and select business objects and services from the PeopleSoft Enterprise server, and to generate business object definitions and related artifacts.

Selecting business objects and services

To specify which objects you want to process, you use the external service wizard.

Before you begin

Make sure that you have set the connection properties for the external service wizard.

About this task

To find and select one or more objects, use the following procedure.

Procedure

1. In the Find Objects in the Enterprise System window, click or expand the object you want to work with.
2. In the Specify the Configuration Properties for 'object' window, specify configuration properties for the object. (This window is available only if you select the **Prompt for additional configuration settings when adding a business object** check box on the Specify the Discovery Properties window.)
   a. Select the field name to use for performing logical deletes.
   b. In the **Value used to indicate a deleted object** field, type the status value used to indicate that an object is no longer active.
   c. Optional: If you want to use alternate keys during RetrieveAll operations, click **Add** and select the key fields you want to use.
   d. Optional: To set additional parameters, click **Advanced**.
   - At the **Preserve objects in parent update operation** box, click **Add** and select the child objects.
   - At the **Fields that represents generated keys in PeopleSoft** box, click **Add** and select the fields.
   - Click the **Effective date configuration** section, and select from the following effective date options:
     - **Retrieve only current effective dated item**: The corresponding ASI is GetCurrentItem.
     - **Insert at oldest effective date position**: The corresponding ASI is InsAtOldestEffDtPos.
     - **Insert at current effective date position**: The corresponding ASI is InsAtCurrentEffDtPos.
     - **Ignore effective sequence number**: The corresponding ASI is IgnoreEffectiveSequence.
     - **Field that represents effective date value**: The corresponding ASI is EffectiveDate.
     - **Field that represents effective date sequence**: The corresponding ASI is EffectiveSequence.
   - Click **OK**.
3. Click **Next**.

Results

You selected one or more objects to work with.

What to do next

From the Specify Composite Properties window, add or remove associated operations.

Configuring the selected objects

To configure the business object, you specify information about the object (such as, the operation associated with the object).

Before you begin

Make sure that you selected and imported the object.

About this task

To configure the business object, use the following procedure.

Procedure

1. In the Specify Composite Properties window, enter the following information.
   a. Add or remove operations for the object that is added to the service interface.
   b. If you do not want to specify a namespace value, leave the default value for the **Business object namespace field.**
   c. Type a directory location to store the generated business objects.
   d. Optional: To indicate where the generated business objects is stored, type the relative path to the folder in the **Folder** field.

      Note: If you are creating multiple adapter artifacts within a module, ensure that you specify different business object folders for each adapter within the module. For example, if you are creating artifacts for Oracle, JDBC, SAP, and JDE within a module, you need to create different relative folders for each of these adapters. If you do not specify different relative folders, the existing artifacts are overwritten when you generate new artifacts.

   e. If you want the business object contained within a business graph, leave the check box selected.
2. Click Next.

Results

The Specify the Service Generation and Deployment Properties window is displayed.

What to do next

Generate a module that includes the adapter and the business objects.

Setting deployment properties and generating the service

After you select and configure business objects for your module, you can use the external service wizard to configure properties that the adapter uses to connect to a specific enterprise information system. The wizard creates a business integration module where all the artifacts and property values are saved.

Before you begin

Ensure that you configured the business object.

About this task

Use the Specify the Service Generation and Deployment Properties and Specify the Location Properties windows of the external service wizard to perform this task. The connection properties in this task are initialized to the values that the wizard used to connect to the enterprise information system. To configure the module to use other values, change the values here.

Procedure

1. In the Specify the Service Generation and Deployment Properties window, click **Edit Operations** if you want to review or modify the names, or add a description for the operations to the business objects that you are creating.

Figure 6. Specify the Service Generation and Deployment Configuration window

a. In the Edit Operation Names window, select the operation you want to edit, and then click **Edit**.

b. In the Add/Edit properties window, type a new operation name and description, and then click **Finish**.

2. Specify how you want the adapter to get the user name and password at run time.

- To use a J2C authentication alias, select **Using an existing JAAS alias (recommended)** and type the name of the alias in **J2C Authentication Data Entry**. You can specify an existing authentication alias or create one at any time before deploying the module. The name is case-sensitive and includes the node name.

- To use activation specification properties, select **Using security properties from managed connection factory** and type the values in the **User name** and **Password** fields.

- If you want to administer the user name and password from other mechanism, select **Other**.

3. In the **Deploy connector project** field, specify whether to include the adapter files in the module. Select one of the following values:

- **With module for use by single application**. When the adapter files are embedded in the module, you can deploy the module to any application server. Use an embedded adapter when you have a single module using the adapter or if multiple modules must run different versions of the adapter. You can use an embedded adapter to upgrade the adapter in a single module without the risk of destabilizing other modules by changing their adapter version.

- **On server for use by multiple applications**. If you do not include the adapter files in a module, you must install them as a stand-alone adapter on each application server where you want to run the module. Use a stand-alone adapter when multiple modules can use the same version of the adapter and you want to administer the adapter in a central location. A stand-alone adapter can also reduce the resources required by running a single adapter instance for multiple modules.

4. If you selected **On server for use by multiple applications** in the previous step, indicate how you want to specify the connection properties.

- If you manually created and configured a managed connection factory or activation specification on the server or if you already deployed an application that connects to the same enterprise information system using the same managed connection factory or activation specification properties, you can reuse the managed connection factory or activation specification by specifying the name of its Java Naming and Directory Interface (JNDI) data source. The following figure shows the **JNDI Lookup Name** field.

  a. In **Connection settings**, select **Use predefined connection properties**.

  b. In **JNDI Lookup Name**, type the name of the JNDI data source for an existing managed connection factory or activation specification.

  c. Click **Next** to complete this task.

- If this application is the first that connects to the enterprise information system with a specific user name and password, or if you want to administer the user name and password separately from other applications, select **Specify connection properties**.

5. Review and, if necessary, change the values of the required connection properties. The **Host name**, **Port number**, and **Language code** fields are initialized with the connection information that you specified when you started

the wizard. You can also connect to an alternate enterprise information system, although the schema names must be the same in both. The format of the connection properties is specific to the enterprise information system. For more information about the properties, see "Managed connection factory properties" on page 127.

6. Each of the following areas under **Advanced** can be expanded to reveal additional fields:

   - Processing methods for illegal XML characters
   - Event polling configuration
   - Event delivery configuration
   - Event configuration
   - Additional properties
   - Logging and tracing properties
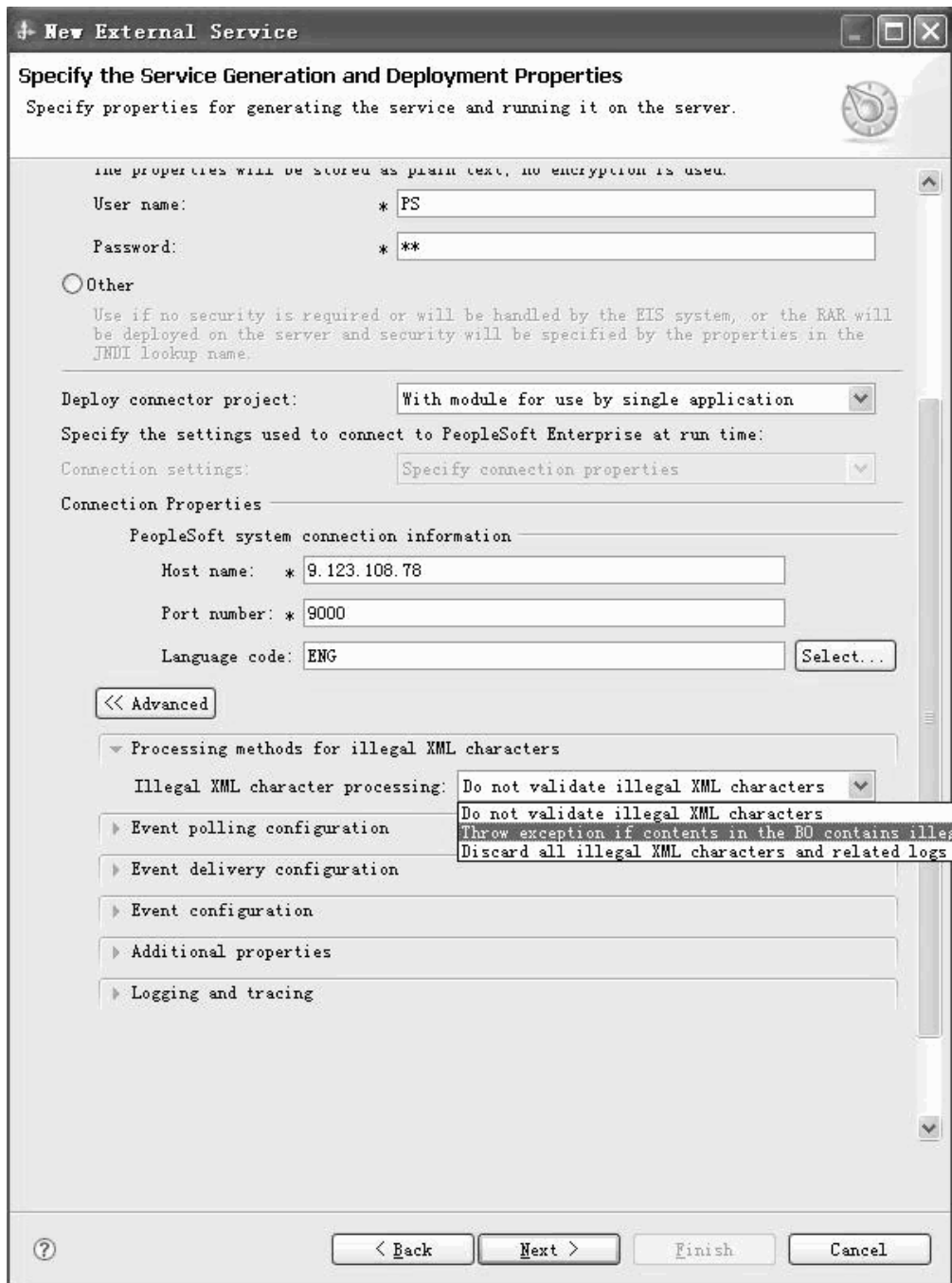
Figure 7. Advanced options for inbound processing

- **Processing methods for illegal XML characters**
    a. From the **Illegal XML character processing** drop-down list, select one of the options that correspond to your requirement:
        – **Do not validate illegal XML characters**, if you want to work with the default adapter behavior.
        – **Throw exception if contents in the BO contains illegal XML characters**, if you want to proceed after you get an exception message, and log the illegal XML characters in the trace file, at runtime.

– **Discard all illegal XML characters and related logs**, if you want the adapter to discard the illegal XML characters, and log the characters in the trace file, at runtime.
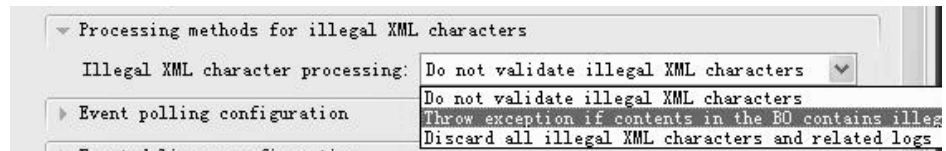
- **Event polling configuration**
  a. In **Interval between polling periods**, type the number of milliseconds that the adapter waits between polling periods. For more information, see "Interval between polling periods (PollPeriod)" on page 149.
  b. In **Maximum events in polling period**, type the number of events to deliver in each polling period. For more information, see "Maximum events in polling period (PollQuantity)" on page 150.
  c. In **Retry interval if connection fails**, type the number of milliseconds to wait before trying to connect after a connection failure during polling. For more information, see "Time between retries in case of system connection failure (RetryInterval)" on page 151.
  d. In **Number of times to retry the system connection**, type the number of times to try the connection again before reporting a polling error. For more information, see "Maximum number of retries in case of system connection failure (RetryLimit)" on page 150.
  e. If you want the adapter to stop if polling errors occur, select **Stop the adapter when an error is encountered while polling**. If you do not select this option, the adapter logs an exception but continues to run. For more information, see "Stop the adapter when an error is encountered while polling (StopPollingOnError)" on page 152.
  f. Optionally, you can select **Retry EIS connection on startup** if you want the adapter to try a failed connection again when starting. For more information, see "Retry EIS connection on startup (RetryConnectionOnStartup)" on page 151.
  g. Select the calendar based scheduling option to create calendar based polling for inbound activities. You can schedule your business activities, when you create a new calendar in IBM Integration Designer. The option of working with the calendar based scheduling feature is only possible with IBM Integration Designer as the tooling environment. The following figure helps you to schedule a calendar polling option.
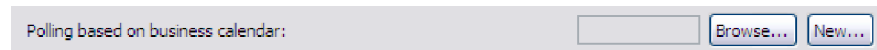
Figure 8. Polling based on business calendar

You can either select a blank calendar or create a new calendar for a module or library.

Note: When you select a blank calendar, you will not be able to set predefined time intervals. You have to define your time intervals. When you create a calendar using a predefined template, you can define time intervals for each template.

– To select an existing calendar for a module or library, click **Browse**. In the Select a Business Calendar window, you can search for existing calendar files (`*cal`) in the IBM Integration Designer workspace.

1) In the **Filter by name** field, type the calendar name or name pattern. The calenders matching the pattern are displayed in the **Matching business calendars** area.

2) Select a calendar and click **OK** to return to the external service wizard.

– To create a new calendar entry for a module or library, click **New**. The Create a business calendar window is displayed.

1) In the **Module or library** field, click **Browse** to select an existing calendar module or click **New** to create a module for the new calendar.

2) In the **Folder** field, click **Browse** to select an existing folder or create a new folder for the calendar.

3) In the **Name** field, enter a name for the new calendar.

- To create a non template calendar, click **Finish**. Or

- To generate the calendar based on a predefined template, click **Next**. In the Use a template window, select the **Create a calendar using one of the templates** check box and click **Finish**.

The new business calendar is created and available in the Business Integration view. Once you complete the wizard, you can view or modify the calendar schedules in the Business Integration view using the Business Calendar Editor. You can modify the intervals and exceptions, or add new entries for these elements. For more details about working with business calendars, see Business calendars.

Note: You must deploy the Business Calendar module to the same IBM Business Process Manager or WebSphere Enterprise Service Bus instance, along with the inbound application. If you do not map these two connections to the same server instance, the inbound application using the business calendar will by default, poll as there is no calendar configured.

- **Event delivery configuration**

   a. In **Type of delivery**, select the delivery method. The methods are described in "Delivery type (DeliveryType)" on page 146.

   b. If you want to ensure that events are delivered only once and to only one export, select **Ensure once-only delivery**. This option might reduce performance but does not result in duplicate or missing event delivery. For more information, see "Ensure once-only event delivery (AssuredOnceDelivery)" on page 144.

   c. By default, the adapter processes all events that it finds when it polls. If you do not want it to process events that have timestamps later than the current time, select the **Do not process events that have a timestamp in the future** check box. For more information, see "Do not process events that have a timestamp in the future (FilterFutureEvents)" on page 146

   d. In **Event types to process**, type a comma-separated list of the business objects for which you want events delivered. Leave this field blank to receive events for all business object types.

For example, if you want to receive events only when the Customer and Order tables, but not other tables, are changed in the enterprise information system, set this field to `Customer,Order`.

For more information, see "Event types to process (EventTypeFilter)" on page 147.

   e. In **Adapter Instance for event filtering**, type a string value that determines whether this adapter instance processes specific events in the event store. For more information, see "Adapter Instance for event filtering (AdapterInstanceEventFilter)" on page 143.

   f. Under **Number of connections for event delivery**, specify the minimum and maximum number of connections to use to deliver events. For more information, see "Minimum connections (MinimumConnections)" on page 149 and "Maximum connections (MaximumConnections)" on page 149.

- **Event configuration**
  - a. In **Component interface name for event store**, specify the component interface the adapter uses for event notification. For more information, see "Component interface name for event notification (EventCIName)" on page 145.
  - b. In **Delimiter for keys in the event store**, specify the delimiter for the object key name-value pair in the event table. For more information, see "Delimiter for keys in the event store (EventKeyDelimiter)" on page 146.
  - c. In **Java date format for event timestamp**, specify the format used to create the event timestamp. For more information, see "Java date format for event timestamp (DateFormat)" on page 149.

- **Additional properties**

  In **Component interface for testing failed connection**, enter the name of the interface. For more information, see "Component interface for testing failed connection (PingCompInterface)" on page 145.

- **Logging and tracing properties**
  - a. If you have multiple instances of the adapter, set Adapter ID to a value that is unique for this instance. For more information, see "Adapter ID (adapterID)" on page 123.
  - b. Select **Disguise user data as 'XXX' in log and trace files** if you want to prevent sensitive user data from being written to log and trace files. For more information, see "Disguise user data as "XXX" in log and trace files (hideConfidentialTrace) " on page 129.

7. Click **Next**. The Specify the Location Properties window is displayed.
8. In the Specify the Location Properties window, create a module.
   - a. Click **New** in the **Module** field.
   - b. Select a Business Integration Project Type window, select **Create a module project** or **Create a mediation module project**, then click Next.
   - c. In the Create a Module window, type a name for the module.
   - d. Specify whether you want to open the module in the assembly diagram (for module projects) or whether you want to create a mediation flow component (for mediation module projects). By default, these choices are selected.
   - e. Click **Finish**. The new module is created. When the creation process is finished, the Create a Module window closes, and the new module appears in the Module list in the Specify the Location Properties window.

Results

The new module is added to the Business Integration perspective.

What to do next

Export the module as an EAR file for deployment.

Chapter 5. Modifying artifacts

Often business requirements mandate changes to the backend enterprise information system (EIS) data structures. These changes call for regeneration and reconfiguration of artifacts (import and export) that were previously generated using the external service wizard.

A few business scenarios where the output from one enterprise service discovery flow can be reused in successive flows, are outlined as follows:

- When you want to add a new object to the object set.
- When you want to modify configurations on selected objects like changes to operations, operation names, and service level settings like security, transactions, and reliability.
- When you want to remove a discovered object from the object set.
- When you want to rediscover existing objects in the service to synchronize the service if the objects in the back-end system have been updated.

To modify existing artifacts, you can invoke the wizard in one of the following ways. The external service wizard is initialized with previously configured settings.

- In the assembly editor, select the component you want to modify, right-click, and select **Edit Binding**.
- In the Business Integration view, select the component you want to modify, right-click, and select **Edit Binding**.
- Select the component in the assembly editor and select the Properties view. In the Binding tab, click the **Edit** link.

Note: The Edit Binding option is available for artifacts generated using IBM Integration Designer 7.0 only. If you are importing a project interchange from an earlier version of IBM Integration Designer, the Edit Binding option is not available. If you have made any manual changes to the configuration, running the wizard again will overwrite these changes.

Modifying service import

Modify an import component by rediscovering and reconfiguring the objects using the Edit Binding option in IBM Integration Designer.

About this task

You can invoke the external service wizard to modify the information of a service import interface. The wizard automatically populates the existing information for the selected import interface. You can modify the objects and services, and then regenerate the import component with the modified data.

Procedure

1. Invoke the external service wizard for the selected service interface import component using one of the following methods.
    - In the assembly editor, select the component you want to modify, right-click and select **Edit Binding**.
    - In the Business Integration view, select the component you want to modify, right-click and select **Edit Binding**.

- Select the interface in the assembly editor and select the Properties view. In the Binding tab, click the **Edit** link.

  The Find Objects in the Enterprise System window of the external service wizard is displayed. The external service wizard automatically populates the existing configuration details for the selected import interface.

2. In the Find Objects in the Enterprise System window, make the required changes. For more information about discovering objects, see "Selecting business objects and services" on page 40.

   Note: To change the connection properties for the external service wizard, click **Back** and change the properties in the Specify the Discovery Properties window. For more information see, "Setting connection properties for the external service wizard" on page 39.

   a. To modify the configuration of an object from the **Selected objects** list, select the object name and then click the 🖉 (Edit) icon.

   b. To remove an object from the **Selected objects** list, select the object name and then click the [<] (Remove) button.

3. Click **Next**. If you click **Cancel**, the changes you made in the previous step does not take effect.

4. In the Specify Composite Properties window, specify properties that apply to all business objects. For more information, see "Configuring the selected objects" on page 42.

5. Click **Next**.

6. In the Service Generation window, modify the service operations if required.

7. Click **Finish**.

Results

The artifacts are regenerated.

What to do next

You can test and deploy your module.

Modifying service export

Modify an export component by rediscovering and reconfiguring the objects using the Edit Binding option in IBM Integration Designer.

About this task

You can invoke the external service wizard to modify the information of a service export interface. The wizard automatically populates the existing information for the selected export interface. You can modify the objects and services, and then regenerate the export component with the modified data.

Procedure

1. Invoke the external service wizard for the selected service interface export component using one of the following methods.

   - In the assembly editor, select the component you want to modify, right-click and select **Edit Binding**.

- In the Business Integration view, select the component you want to modify, right-click and select **Edit Binding**.
- Select the interface in the assembly editor and select the Properties view. In the Binding tab, click the **Edit** link.

The Find Objects in the Enterprise System window of the external service wizard is displayed. The external service wizard automatically populates the existing configuration details for the selected export interface.

2. In the Find Objects in the Enterprise System window, make the required changes. For more information about discovering objects, see "Selecting business objects and services" on page 48.

   Note: To change the connection properties for the external service wizard, click **Back** and change the properties in the Specify the Discovery Properties window. For more information see, "Setting connection properties for the external service wizard" on page 39.

   a. To modify the configuration of an object from the **Selected objects** list, select the object name and then click the ✎ (Edit) icon.

   b. To remove an object from the **Selected objects** list, select the object name and then click the ⬚ (Remove) button.

3. Click **Next**. If you click **Cancel**, the changes you made in the previous step does not take effect.

4. In the Specify Composite Properties window, specify properties that apply to all business objects. For more information, see "Configuring the selected objects" on page 50.

5. Click **Next**.

6. In the Service Generation window, modify the service operations if required.

7. Click **Finish**.

Results

The artifacts are regenerated.

What to do next

You can test and deploy your module.

Chapter 6. Changing interaction specification properties

To change interaction specification properties for your adapter module after generating the service, use the assembly editor in IBM Integration Designer.

Before you begin

You must have used the external service wizard to generate a service for the adapter.

About this task

You might want to change interaction specification properties after you have generated a service for the adapter. Interaction specification properties, which are optional, are set at the method level, for a specific operation on a specific business object. The values you specify appear as defaults in all parent business objects generated by the external service wizard. You can change these properties before you export the EAR file. You cannot change these properties after you deploy the application.

To change the interaction specification properties, use the following procedure:

Procedure

1. From the Business Integration perspective of IBM Integration Designer, expand the module name.
2. Expand **Assembly Diagram** and double-click the interface.
3. Click the interface in the assembly editor. The module properties are displayed.
4. Click the **Properties** tab. You can also right-click the interface in the assembly diagram and click **Show in Properties**.
5. Under **Binding**, click **Method bindings**. The methods for the interface are displayed, one for each combination of business object and operation.
6. Select the method whose interaction specification property you want to change.
7. Change the property in the **Generic** tab. Repeat this step for each method whose interaction specification property you want to change.

Results

The interaction specification properties associated with your adapter module are changed.

What to do next

Deploy the module.

Chapter 7. Deploying the module

Deploy a module to place the files that make up your module and adapter into an operational environment for production or testing. In IBM Integration Designer, the integrated test environment features runtime support for IBM Business Process Manager or WebSphere Enterprise Service Bus, or both, depending on the test environment profiles that you selected during installation.

Deployment environments

There are test and production environments into which you can deploy modules and adapters.

In IBM Integration Designer, you can deploy your modules to one or more servers in the test environment. This is typically the most common practice for running and testing business integration modules. However, you can also export modules for server deployment on IBM Business Process Manager or WebSphere Enterprise Service Bus as EAR files using the administrative console or command-line tools.

Deploying the module for testing

In IBM Integration Designer, you can deploy a module that includes an embedded adapter to the test environment and work with server tools that enable you to perform such tasks as editing server configurations, starting, and stopping servers and testing the module code for errors. The testing is generally performed on the interface operations of your components, which enables you to determine whether the components are correctly implemented and the references are correctly wired.

Adding external software dependencies on the server

The adapter needs its specific JAR files installed on the IBM Business Process Manager or WebSphere Enterprise Service Bus server to be able to communicate with the PeopleSoft Enterprise server.

Before you begin

To obtain the required files and copy them to IBM Business Process Manager or WebSphere Enterprise Service Bus, use the following procedure.

Procedure

1. Obtain the psjoa.jar dependency JAR file from the workstation on which the PeopleSoft server is installed and generate the other Java API JAR file for Component Interfaces using the PeopleSoft tools. Depending on the version of PeopleSoft Enterprise you use, the software dependencies differ.

   Note: See the PeopleSoft documentation for more information about using the PeopleSoft tools.

2. Copy the files to the server.
   - In a testing environment in IBM Integration Designer, copy the files to the ${WAS_INSTALL_ROOT}/runtimes/bi_v7/lib/ext directory.

- In a production environment, copy the files to the ${WAS_INSTALL_ROOT}/ lib/ext directory of IBM Business Process Manager or WebSphere Enterprise Service Bus.

Generating and wiring a target component for testing inbound processing

Before deploying to the test environment a module that includes an adapter for inbound processing, you must first generate and wire a target component. This target component serves as the *destination* to which the adapter sends events.

Before you begin

You must have generated an export module, using the external service wizard.

About this task

Generating and wiring a target component for inbound processing is required in a testing environment only. It is not necessary when deploying the adapter in a production environment.

The target component receives events. You *wire* the export to the target component (connecting the two components) using the assembly editor in IBM Integration Designer. The adapter uses the wire to pass event data (from the export to the target component).

Procedure

1. Create the target component.
   a. From the Business Integration perspective of IBM Integration Designer, expand **Assembly Diagram** and double-click the export component. If you did not change the default value, the name of the export component is the name of your adapter + **InboundInterface**.

      An interface specifies the operations that can be called and the data that is passed, such as input arguments, returned values, and exceptions. The **InboundInterface** contains the operations required by the adapter to support inbound processing and is created when you run the external service wizard.
   b. Create a new component by expanding **Components**, selecting **Untyped Component**, and dragging the component to the Assembly Diagram.

      The cursor changes to the placement icon.
   c. Click the component to have it displayed in the Assembly Diagram.
2. Wire the components.
   a. Click and drag the export component to the new component.
   b. Save the assembly diagram. Click **File** > **Save**.
3. Generate an implementation for the new component.
   a. Right-click on the new component and select **Generate Implementation** > **Java**.
   b. Select **(default package)** and click **OK**. This creates an endpoint for the inbound module.

      The Java implementation is displayed in a separate tab.
   c. **Optional:** Add print statements to print the data object received at the endpoint for each of the endpoint methods.

d. Click **File** > **Save** to save the changes.

What to do next

Continue deploying the module for testing.

Adding the module to the server

In IBM Integration Designer, you can add modules to one or more servers in the test environment.

Before you begin

If the module you are testing uses an adapter to perform inbound processing, generate and wire a *target component* to which the adapter sends the events.

About this task

In order to test your module and its use of the adapter, you need to add the module to the server.

Procedure

1. *Conditional:* If there are no servers in the **Servers** view, add and define a new server by performing the following steps:
   a. Place your cursor in the **Servers** view, right-click, and select **New** > **Server**.
   b. From the Define a New Server window, select the server type.
   c. Configure servers settings.
   d. Click **Finish** to publish the server.
2. Add the module to the server.
   a. Switch to the servers view. In IBM Integration Designer, select **Windows** > **Show View** > **Servers**.
   a. Start the server. In the **Servers** tab in the lower-right pane of the IBM Integration Designer screen, right-click the server, and then select **Start**.
3. When the server status is *Started*, right-click the server, and select **Add and Remove Projects**.
4. In the Add and Remove Projects screen, select your project and click **Add**. The project moves from the **Available projects** list to the **Configured projects** list.
5. Click **Finish**. This deploys the module on the server.
   The Console tab in the lower-right pane displays a log while the module is being added to the server.

What to do next

Test the functionality of your module and the adapter.

Testing the module for outbound processing using the test client

Test the assembled module and adapter for outbound processing using the IBM Integration Designer integration test client.

Before you begin

You need to add the module to the server first.

About this task

Testing a module is performed on the interface operations of your components, which enables you to determine whether the components are correctly implemented and the references are correctly wired.

Procedure

1. Select the module you want to test, right-click on it, and select **Test** > **Test Module**.
2. For information about testing a module using the test client, see the *Testing modules and components* topic in the IBM Integration Designer information center.

What to do next

If you are satisfied with the results of testing your module and adapter, you can deploy the module and adapter to the production environment.

Deploying the module for production

Deploying a module created with the external service wizard to IBM Business Process Manager or WebSphere Enterprise Service Bus in a production environment is a two-step process. First, you export the module in IBM Integration Designer as an enterprise archive (EAR) file. Second, you deploy the EAR file using the IBM Business Process Manager or WebSphere Enterprise Service Bus administrative console.

Adding external software dependencies on the server

The adapter needs its specific JAR files installed on the IBM Business Process Manager or WebSphere Enterprise Service Bus server to be able to communicate with the PeopleSoft Enterprise server.

Before you begin

To obtain the required files and copy them to IBM Business Process Manager or WebSphere Enterprise Service Bus, use the following procedure.

Procedure

1. Obtain the psjoa.jar dependency JAR file from the workstation on which the PeopleSoft server is installed and generate the other Java API JAR file for Component Interfaces using the PeopleSoft tools. Depending on the version of PeopleSoft Enterprise you use, the software dependencies differ.

   Note: See the PeopleSoft documentation for more information about using the PeopleSoft tools.

2. Copy the files to the server.
   - In a testing environment in IBM Integration Designer, copy the files to the ${WAS_INSTALL_ROOT}/runtimes/bi_v7/lib/ext directory.
   - In a production environment, copy the files to the ${WAS_INSTALL_ROOT}/lib/ext directory of IBM Business Process Manager or WebSphere Enterprise Service Bus.

Installing the RAR file (for modules using stand-alone adapters only)

If you chose not to embed the adapter with your module, but instead choose to make the adapter available to all deployed applications in the server instance, you need to install the adapter in the form of a RAR file to the application server. A RAR file is a Java archive (JAR) file that is used to package a resource adapter for the Java EE Connector Architecture (JCA).

Before you begin

You must set **Deploy connector project** to **On server for use by multiple adapters** in the Specify the Service Generation and Deployment Properties window of the external service wizard.

About this task

Installing the adapter in the form of a RAR file results in the adapter being available to all Java EE application components running in the server run time.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Resources** > **Resource Adapters** > **Resource adapters**.
5. In the Resource adapters page, click **Install RAR**.
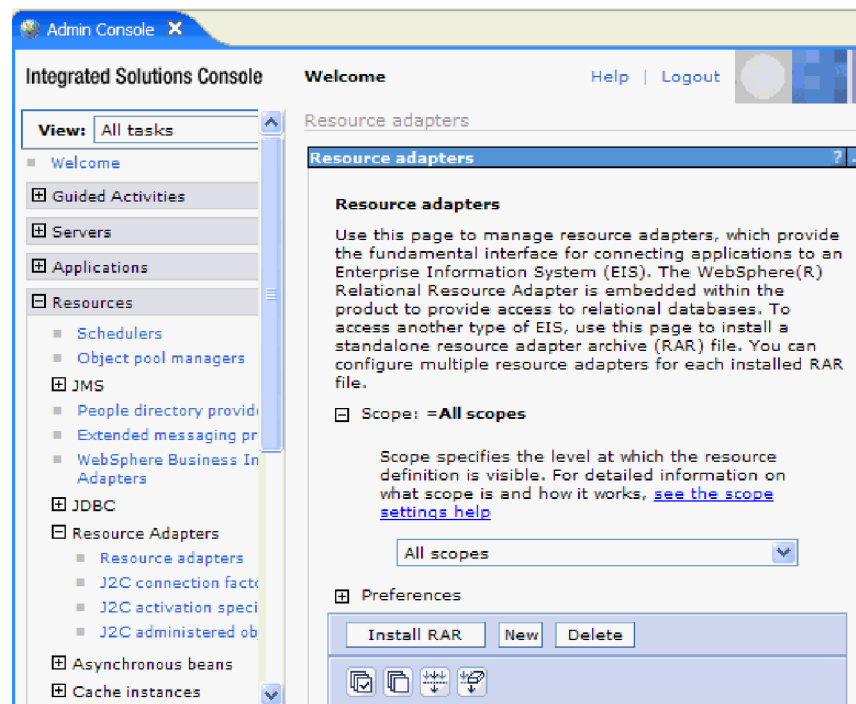
Figure 9. The Install RAR button on the Resource adapters page

6. In the Install RAR file page, click **Browse** and navigate to the RAR file for your adapter.

   The RAR files are typically installed in the following path:
   IID_installation_directory/ResourceAdapters/*adapter_name*/adapter.rar

7. Click **Next**.

8. Optional: In the Resource adapters page, change the name of the adapter and add a description.

9. Click **OK**.

10. Click **Save** in the **Messages** box at the top of the page.

What to do next

The next step is to export the module as an EAR file that you can deploy on the server.

Exporting the module as an EAR file

Using IBM Integration Designer, export your module as an EAR file. By creating an EAR file, you capture all of the contents of your module in a format that can be easily deployed to IBM Business Process Manager or WebSphere Enterprise Service Bus.

Before you begin

Before you can export a module as an EAR file, you must have created a module to communicate with your service. The module should be displayed in the IBM Integration Designer Business Integration perspective.

About this task

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

Procedure

1. Right-click the module and select **Export**.

2. In the Select window, expand **Java EE**.

3. Select **EAR file** and click **Next**.

4. Optional: Select the correct EAR application. The EAR application is named after your module, but with "App" added to the end of the name.

5. Browse for the folder on the local file system where the EAR file will be placed.

6. Optional: To export the source files, select the **Export source files** check box. This option is provided in case you want to export the source files in addition to the EAR file. Source files include files associated with Java components, data maps, and so on.

7. Optional: To overwrite an existing file, click **Overwrite existing file**.

8. Click **Finish**.

Results

The contents of the module are exported as an EAR file.

What to do next

Install the module in the administrative console. This deploys the module to IBM Business Process Manager or WebSphere Enterprise Service Bus.

Installing the EAR file

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

Before you begin

You must have exported your module as an EAR file before you can install it on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

To install the EAR file, perform the following procedure. For more information about clustering adapter module applications, see the http://www.ibm.com/software/webservers/appserv/was/library/.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Applications** > **New Application** > **New Enterprise Application**.

Figure 10. Preparing for the application installation window

5. Click **Browse** to locate your EAR file and click **Next**. The EAR file name is the name of the module followed by "App."

6. Optional: If you are deploying to a clustered environment, complete the following steps.
   a. On the **Step 2: Map modules to servers** window, select the module and click **Next**.
   b. Select the name of the server cluster.
   c. Click **Apply**.
7. Click **Next**. In the Summary page, verify the settings and click **Finish**.
8. Optional: If you are using an authentication alias, complete the following steps:
   a. Expand **Security** and select **Business Integration Security**.
   b. Select the authentication alias that you want to configure. You must have administrator or operator rights to change the authentication alias configurations.
   c. Optional: If it is not already specified, type the **User name**.
   d. If it is not already specified, type the **Password**.
   e. If it is not already specified, type the password again in the **Confirm Password** field.
   f. Click **OK**.

Results

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

What to do next

If you want to set or reset any properties or you would like to cluster adapter project applications, make those changes using the administrative console before configuring troubleshooting tools.

Deploying the module in a clustered environment

In IBM Integration Designer, you can deploy the IBM WebSphere Adapter for PeopleSoft Enterprise in a clustered environment.

To deploy the module in a clustered environment, use any of the following approaches.

- **Embedded module:** The adapter is embedded in the application and specific to it. The adapter cannot be shared between multiple applications.
- **Node level module with embedded activation specification:** The adapter is deployed at the node level, with the activation specification created during module creation. The adapter can be shared across multiple applications.
- **Node level module with JNDI activation specification reference:** The adapter is deployed at the node level, and the application provides a JNDI reference to the activation specification. You must create the reference at the cluster scope from the administrative console, with the same JNDI name. The adapter can be shared across multiple applications.

Deploying module embedded in the application

The adapter is deployed embedded in the application and specific to it. The adapter cannot be shared between multiple applications.

Before you begin

The following steps are a necessary prerequisite to configure and deploy the module.

- IBM Integration Designer version 7.5.0.0 or above.
- A clustered topology deployment environment on the IBM Business Process Manager or WebSphere Enterprise Service Bus available from IBM Integration Designer.
- Create a clustered topology deployment environment, as shown in the following **Gold Topology** configuration figure.

Figure 11. Deployment environment

- Deploy the adapter and the adapter applications (EAR files) in the AppTarget (the target that hosts the SCA container).
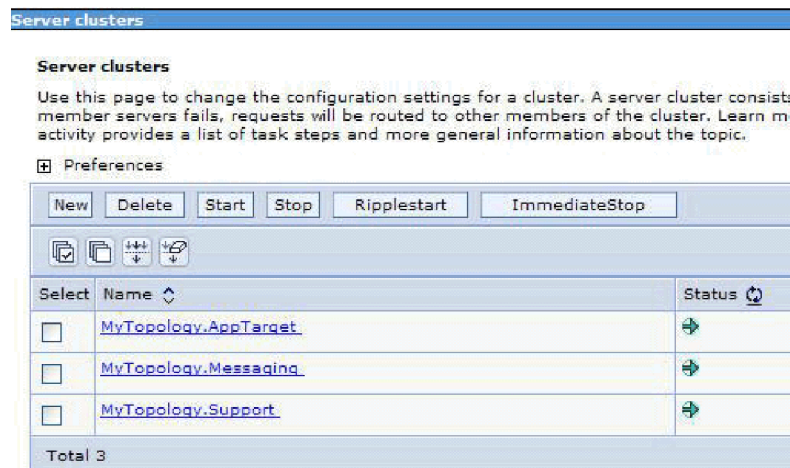
Figure 12. Deployment environment showing server clusters

About this task

To create an application with the embedded adapter, use the external service wizard.

Procedure

1. In the Service Configuration Properties window, from the **Deploy connector project** property list, select **With module for use by single application**.
2. Create the module as described in the Business process management samples for WebSphere Adapters.
3. In the **Dependencies** option for the module, after the module is created, ensure that the **Deploy with module** option is selected for the adapter.
4. If the server is not running, right-click your server in the **Servers** view and select **Start**.
5. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
6. From the **Deployment Manager Admin Console**, click **Install applications** to deploy the application.
7. On the **Step 2: Map modules to servers** window, select the module and click **Next**. For the embedded adapter option, the adapter is deployed as part of the application.
8. In the Enterprise Applications view, select the new application *<adapter_name>***EmbeddedModuleApp**. The new application is displayed after the application is deployed at the deployment manager level.
9. Select the node and click **Installed applications** to view the deployed application on each individual node.

Results

The resource adapter is embedded and deployed as part of the application.

Deploying module at node level with embedded activation specification

The adapter is deployed at the node level, with the activation specification created during module creation. The adapter can be shared across multiple applications.

Before you begin

The following steps are a necessary prerequisite to configure and deploy the module.

- IBM Integration Designer version 7.5.0.0 or above.
- A clustered topology deployment environment on the IBM Business Process Manager or WebSphere Enterprise Service Bus available from IBM Integration Designer.
- Create a clustered topology deployment environment, as shown in the following **Gold Topology** configuration figure.

Figure 13. Deployment environment

- Deploy the adapter and the adapter applications (EAR files) in the AppTarget (the target that hosts the SCA container).

Figure 14. Deployment environment showing server clusters

About this task

To create an application with the node level adapter and activation specification properties specified in the module itself, use the external service wizard.

Procedure

1. In the Service Configuration Properties window, from the **Deploy connector project** property list, select **On server for use by multiple applications**.
2. From the **Connection properties** list, select **Use properties below**.
3. Create the module as described in the Business process management samples for WebSphere Adapters.
4. In the **Dependencies** option for the module, ensure that the **Deploy with module** option is not selected for the adapter. Here, the adapter is not part of the module, therefore you must deploy the adapter before deploying the application.
5. If the server is not running, right-click your server in the **Servers** view and select **Start**.

6. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**. Log on to the administrative console.

7. To deploy the adapter at individual nodes, click **Resources** > **Resource Adapters** > **Resource adapters**. In the clustered environment, you must install the adapter in each node separately.

8. In the Resource adapters page, click **Install RAR**.

9. In the Install RAR file page, click **Browse** and navigate to the RAR file for your adapter. Deploy the RAR on each node.

   The RAR files are typically installed in the following path: *IID_installation_directory*/ResourceAdapters/*adapter_name*/adapter.rar

10. For deployment at node level, do not select any **Scope** because the scope is always **Node**. Click **Next**.

11. Optional: In the Resource adapters page, change the name of the adapter and add a description. Click **OK**.

12. Click **Save** in the **Messages** box at the top of the page.

13. For node level deployment, check if the adapter RAR is deployed at the node level.

14. To deploy the adapter at the cluster level, click **Resources** > **Resource Adapters** > **Resource adapters**.

15. In the Resource adapters window, set the **Scope** to **Cluster**, and then click **New**.

16. Select the RAR deployed at the node level.

17. Check if the adapter RAR is now deployed at the cluster level. Deploy the application after the adapter is deployed at the node level on the individual nodes, and then at the cluster level.

18. From the **Deployment Manager Admin Console**, click **Install applications** to deploy the application.

19. On the **Step 2: Map modules to servers** window, select the module and click **Next**. The adapter is not part of the deployed application.

20. In the **Admin Console**, click **Resources** > **Resource Adapters** > **IBM WebSphere Adapter for PeopleSoft Enterprise** > **J2C activation specifications** to view the activation specification from the adapter deployed at the cluster level.

Results

The resource adapter is deployed at the node level, with the activation specification.

Deploying module at node level with JNDI activation specification

The adapter is deployed at the node level, and the application provides a JNDI reference to the activation specification. You must create the activation specification with the same JNDI name at the cluster scope from the administrative console. The adapter can be shared across multiple applications

Before you begin

The following steps are a necessary prerequisite to configure and deploy the module.

- IBM Integration Designer version 7.5.0.0 or above.
- A clustered topology deployment environment on the IBM Business Process Manager or WebSphere Enterprise Service Bus available from IBM Integration Designer.
- Create a clustered topology deployment environment, as shown in the following **Gold Topology** configuration figure.
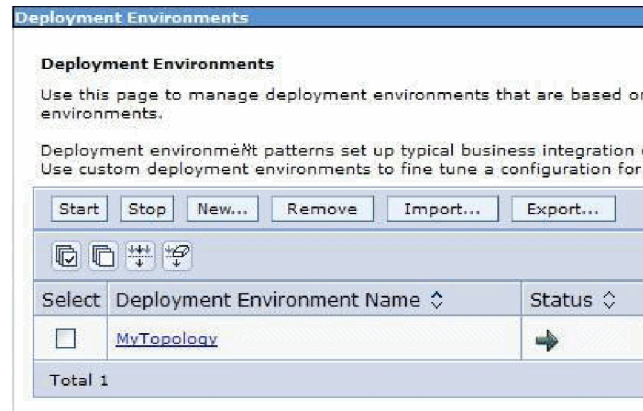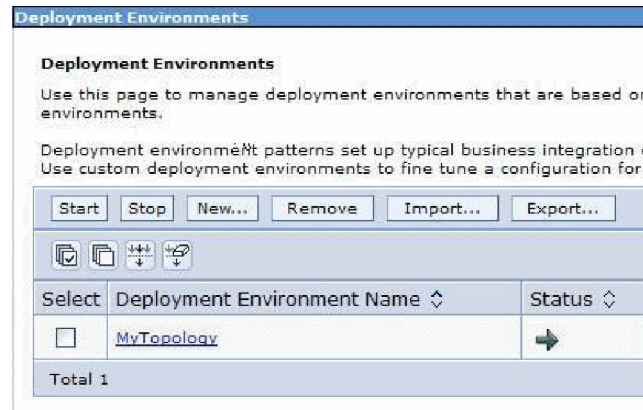

Figure 15. Deployment environment

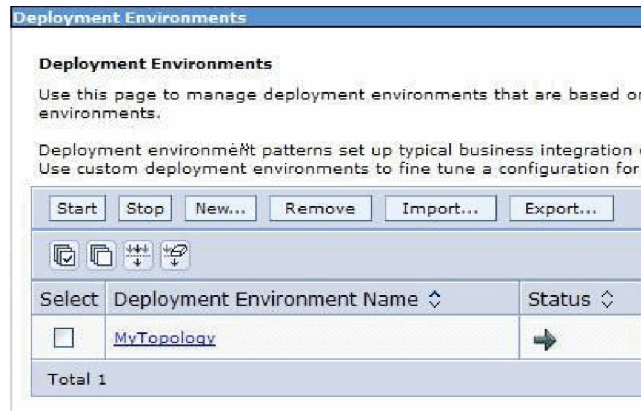- Deploy the adapter and the adapter applications (EAR files) in the AppTarget (the target that hosts the SCA container).
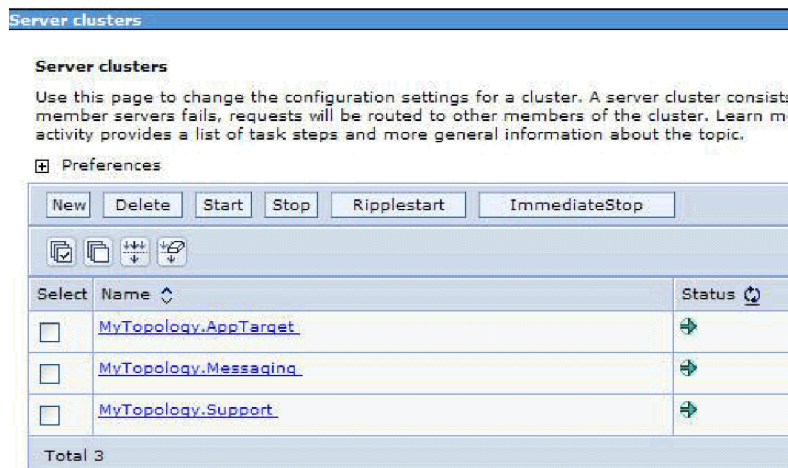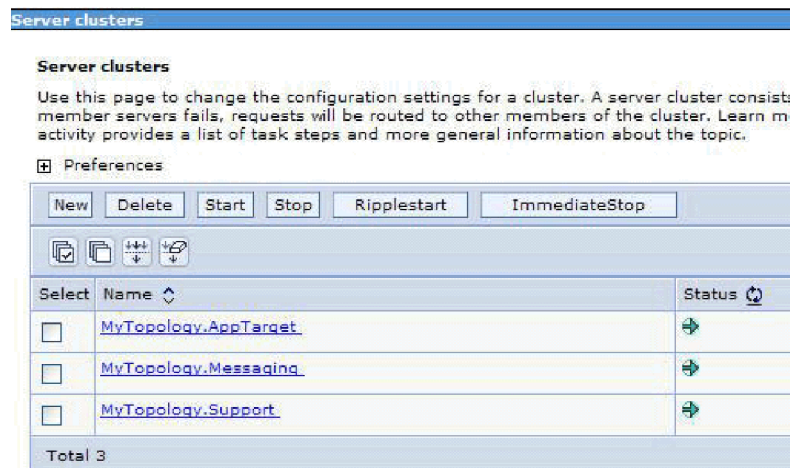

Figure 16. Deployment environment showing server clusters

About this task

To create an application with the node level adapter and activation specification properties specified in the module itself, use the external service wizard.

Procedure

1. In the Service Configuration Properties window, from the **Deploy connector project** property list, select **On server for use by multiple applications**.
2. From the **Connection properties** list, select **Use JNDI lookup name configured on server**.

3. In the **JNDI lookup name** property field, specify the JNDI name. Use this same JNDI name when you create the activation specification from the Admin Console.

4. Create the module as described in the Business process management samples for WebSphere Adapters.

5. In the **Dependencies** option for the module, ensure that the **Deploy with module** option is not selected for the adapter.

6. If the server is not running, right-click your server in the **Servers** view and select **Start**.

7. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**. Log on to the administrative console.

8. To install the adapter at the node level, click **Resources** > **Resource Adapters** > **Resource adapters**. In the clustered environment, you must install the adapter in each node separately.

9. In the Resource adapters page, click **Install RAR**.

10. In the Install RAR file page, click **Browse** and navigate to the RAR file for your adapter. Deploy the RAR on each node.

    The RAR files are typically installed in the following path: *IID_installation_directory*/ResourceAdapters/*adapter_name*/adapter.rar

11. For deployment at node level, do not select any **Scope** because the scope is always **Node**. Click **Next**.

12. Optional: In the Resource adapters page, change the name of the adapter and add a description. Click **OK**.

13. Click **Save** in the **Messages** box at the top of the page.

14. To install the RAR at the cluster level, click **Resources** > **Resource Adapters** > **Resource adapters**

15. In the Resource adapters page, set the **Scope** to **Cluster**, and then click **New**.

16. Select the RAR deployed at the node level, and then check if the adapter RAR is now deployed at the cluster level. Deploy the application after the adapter is deployed at the node level on the individual nodes, and then at the cluster level.

17. From the **Deployment Manager Admin Console**, click **Install applications** to deploy the application.

18. In the **Admin Console**, click **Resources** > **Resource Adapters** > **IBM WebSphere Adapter for PeopleSoft Enterprise** > **J2C activation specifications** > **New** to create the activation specification from the adapter deployed at the cluster level.

19. When installing the adapter, in the **Name** field, you must enter the same name as defined in the RAR.

20. In the **JNDI name** field, you must enter the same name as given during the module creation.

21. Click **Resources** > **Resource Adapters** > **IBM WebSphere Adapter for PeopleSoft Enterprise** > **J2C activation specifications** to check if the JNDI reference on the adapter is same as the one specified for the module.

22. Click **Resources** > **Resource Adapters** > **IBM WebSphere Adapter for PeopleSoft Enterprise** > **J2C activation specifications** > **Custom properties** to set values for the activation specification in the Admin Console.

23. From the **Deployment Manager Admin console**, click **Install applications** to deploy the application after you deploy the RAR and create the activation specification.

24. On the **Step 2: Map modules to servers** page, select the module and click **Next**. The adapter is not part of the deployed application.

Results

The resource adapter is deployed at the node level, with the JNDI activation specification reference.

Chapter 8. Administering the adapter module

When you are running the adapter in a stand-alone deployment, use the administrative console of the server to start, stop, monitor, and troubleshoot the adapter module. In an application that uses an embedded adapter, the adapter module starts or stops when the application is started or stopped.

Configuring logging and tracing

Configure logging and tracing to suit your requirements. Enable logging for the adapter to control the status of event processing. Change the adapter log and trace file names to separate them from other log and trace files.

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 IBM 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.

Note: For more information about monitoring on a IBM Process Server, including service components and event points, see http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5m1/topic/com.ibm.wbpm.admin.doc/topics/welcome_wps_mon.html.

You can change the log configuration statically or dynamically. Static configuration takes effect when you start or restart the application server. Dynamic or run time 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 child logs, which recursively propagate the change to their child log, as necessary.

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

Procedure

1. In the navigation pane of the administrative console, select **Servers** > **WebSphere application servers**.
2. Click the name of the server that you want to work with.
3. Under **Troubleshooting**, click **Logging and tracing**.
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. Click the names of 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.***.
   - For the WebSphere Adapter for PeopleSoft Enterprise only, select the **com.ibm.j2ca.peoplesoft.*** package.
7. 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.
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. Optional: To have static configuration changes take effect, stop and then restart the IBM Process Server.

Results

Log entries from this point forward contain the specified level of information for the selected adapter components.

Changing the log and trace file names

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

Before you begin

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

About this task

You can change the log and trace file names statically or dynamically. Static changes take effect when you start or restart the application server. Dynamic or run time 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.

Procedure

1. In the navigation pane of the administrative console, select **Applications >Application Types>WebSphere enterprise applications**.
2. In the Enterprise Applications list, click the name of the adapter application. This is the name of the EAR file for the adapter, but without the ear file extension. For example, if the EAR file is named Accounting_OutboundApp.ear, then click **Accounting_OutboundApp**.
3. In the Configuration tab, select **Modules>Manage Modules**.
4. In the list of modules, click IBM WebSphere Adapter for PeopleSoft Enterprise.
5. In the Configuration tab, under Additional Properties, click **Resource Adapter**.
6. In the Configuration tab, under Additional Properties, click **Custom properties**.
7. In the Custom Properties table, change the file names.
   a. Click either **logFilename** to change the name of the log file or **traceFilename** to change the name of the trace file.
   b. In the Configuration tab, type the new name in the **Value** field. By default, the log file is called SystemOut.log and the trace file is called trace.log.
   c. Click **Apply** or **OK**. Your changes are saved on your local machine.
   d. To save your changes to the master configuration on the server, use one of the following procedures:
      • **Static change**: Stop and restart the server. This method allows you to make changes, but those changes do not take effect until you stop and start the server.
      • **Dynamic change**: Click the **Save** link in the Messages box above the Custom properties table. Click **Save** again when prompted.

Changing configuration properties for embedded adapters

To change the configuration properties after you deploy the adapter as part of a module, you use the administrative console of the runtime environment. You can update resource adapter properties (used for general adapter operation), managed connection factory properties (used for outbound processing), and activation specification properties (used for inbound processing). For information about configuring logging properties and changing the log and trace file names, see "Configuring logging and tracing" on page 81.

Setting resource adapter properties for embedded adapters

To set resource adapter properties for your adapter after it has been deployed as part of a module, use the administrative console. You select the name of the property you want to configure and then change or set the value.

Before you begin

Your adapter module must be deployed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

Custom properties are default configuration properties shared by all IBM WebSphere Adapters.

To configure properties using the administrative console, use the following procedure:

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Select **Applications** > **Application Types** > **WebSphere enterprise application**.
5. From the Enterprise Applications list, click the name of the adapter module whose properties you want to change. The **Configuration** page is displayed.

Figure 17. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for PeopleSoft Enterprise**.
8. From the **Additional Properties** list, click **Resource Adapter**.
9. On the next page, from the **Additional Properties** list, click **Custom properties**.
10. For each property you want to change, perform the following steps.

    Note: See "Resource adapter properties" on page 122 for more information about these properties.

    a. Click the name of the property. The **Configuration** page for the selected property is displayed.
    b. Change the contents of the **Value** field or type a value, if the field is empty.
    c. Click **OK**.
11. In the Messages area, click **Save**.

Results

The resource adapter properties associated with your adapter module are changed.

Setting managed (J2C) connection factory properties for embedded adapters

To set managed connection factory properties for your adapter after it has been deployed as part of a module, use the administrative console. You select the name of the property you want to configure and then change or set the value.

Before you begin

Your adapter module must be deployed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

You use managed connection factory properties to configure the target PeopleSoft Enterprise server instance.

Note: In the administrative console, the properties are referred to as "J2C connection factory properties."

To configure properties using the administrative console, use the following procedure.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Select **Applications** > **Application Types** > **WebSphere enterprise application**.
5. In the Enterprise Applications list, click the name of the adapter module whose properties you want to change.
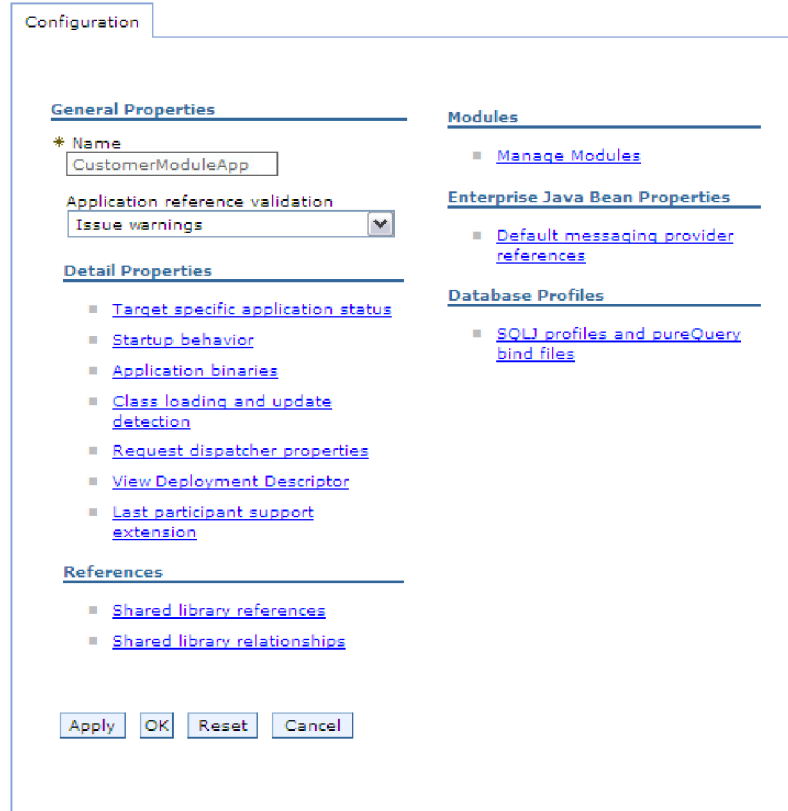
Figure 18. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for PeopleSoft Enterprise**.
8. In the **Additional Properties** list, click **Resource Adapter**.
9. On the next page, from the **Additional Properties** list, click **J2C connection factories**.
10. Click the name of the connection factory associated with your adapter module.
11. In the **Additional Properties** list, click **Custom properties**.

    Custom properties are those J2C connection factory properties that are unique to IBM WebSphere Adapter for PeopleSoft Enterprise. Connection pool and advanced connection factory properties are properties you configure if you are developing your own adapter.
12. For each property you want to change, perform the following steps.

    Note: See "Managed connection factory properties" on page 127 for more information about these properties.
    a. Click the name of the property.
    b. Change the contents of the **Value** field or type a value, if the field is empty.
    c. Click **OK**.
13. In the Messages area, click **Save**.

Results

The managed connection factory properties associated with your adapter module are changed.

Setting activation specification properties for embedded adapters

To set activation specification properties for your adapter after it has been deployed as part of a module, use the administrative console. You select the name of the message endpoint property you want to configure, and then change or set the value.

Before you begin

Your adapter module must be deployed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

You use activation specification properties to configure the endpoint for inbound processing.

To configure properties using the administrative console, use the following procedure.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Select **Applications** > **Application Types** > **WebSphere enterprise application**.
5. From the Enterprise Applications list, click the name of the adapter module whose properties you want to change.
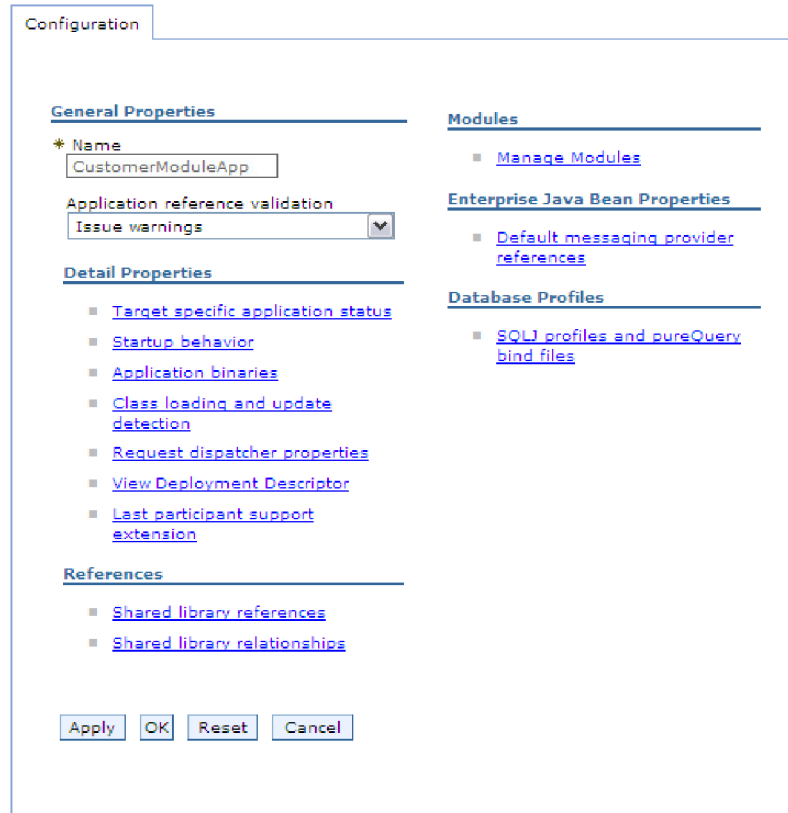
Figure 19. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for PeopleSoft Enterprise**.
8. From the **Additional Properties** list, click **Resource Adapter**.
9. On the next page, from the **Additional Properties** list, click **J2C activation specifications**.
10. Click the name of the activation specification associated with the adapter module.
11. From the **Additional Properties** list, click **J2C activation specification custom properties**.
12. For each property you want to change, perform the following steps.

    Note: See "Activation specification properties" on page 142 for more information about these properties.

    a. Click the name of the property.
    b. Change the contents of the **Value** field or type a value, if the field is empty.
    c. Click **OK**.
13. In the Messages area, click **Save**.

Results

The activation specification properties associated with your adapter module are changed.

Changing configuration properties for stand-alone adapters

To set configuration properties after you install a stand-alone adapter, use the administrative console of the runtime environment. Provide the general information about the adapter and then set the resource adapter properties (which are used for general adapter operation). If the adapter is used for outbound operations, create a connection factory and then set the properties for it. If the adapter is used for inbound operations, create an activation specification and then set the properties for it. For information about configuring logging properties and changing the log and trace file names, see "Configuring logging and tracing" on page 81.

Setting resource adapter properties for stand-alone adapters

To set resource adapter properties for your stand-alone adapter after it has been installed on IBM Business Process Manager or WebSphere Enterprise Service Bus, use the administrative console. You select the name of the property you want to configure and then change or set the value.

Before you begin

Your adapter must be installed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

Custom properties are default configuration properties shared by all IBM WebSphere Adapters.

To configure properties using the administrative console, use the following procedure:

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Resources** > **Resource Adapters** > **Resource adapters**.
5. In the Resource adapters page, click **IBM WebSphere Adapter for PeopleSoft Enterprise**.
6. In the **Additional Properties** list, click **Custom properties**.
7. For each property you want to change, perform the following steps.

   Note: See "Resource adapter properties" on page 122 for more information about these properties.
   a. Click the name of the property.
   b. Change the contents of the **Value** field or type a value, if the field is empty.
   c. Click **OK**.
8. In the Messages area, click **Save**.

Results

The resource adapter properties associated with your adapter are changed.

Setting managed (J2C) connection factory properties for stand-alone adapters

To set managed connection factory properties for your stand-alone adapter after it has been installed on IBM Business Process Manager or WebSphere Enterprise Service Bus, use the administrative console. You select the name of the property you want to configure and then change or set the value.

Before you begin

Your adapter must be installed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

You use managed connection factory properties to configure the target PeopleSoft Enterprise server instance.

Note: In the administrative console, the properties are referred to as "J2C connection factory properties."

To configure properties using the administrative console, use the following procedure:

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Resources** > **Resource Adapters** > **Resource adapters**.
5. In the Resource adapters page, click **IBM WebSphere Adapter for PeopleSoft Enterprise**.
6. In the **Additional Properties** list, click **J2C connection factories**.
7. If you are going to use an existing connection factory, skip ahead to select from the list of existing connection factories.

   Note: If you have selected **Specify connection properties** when you use the external service wizard to configure the adapter module, you do not need to create a connection factory.

   If you are creating a connection factory, perform the following steps:
   a. Click **New**.
   b. In the **General Properties** section of the **Configuration** tab, type a name for the connection factory. For example, you can type `AdapterCF`.
   c. Type a value for **JNDI name**. For example, you can type `com/eis/AdapterCF`.
   d. Optional: Select an authentication alias from the **Component-managed authentication alias** list.
   e. Click **OK**.
   f. In the Messages area, click **Save**.
      The newly created connection factory is displayed.
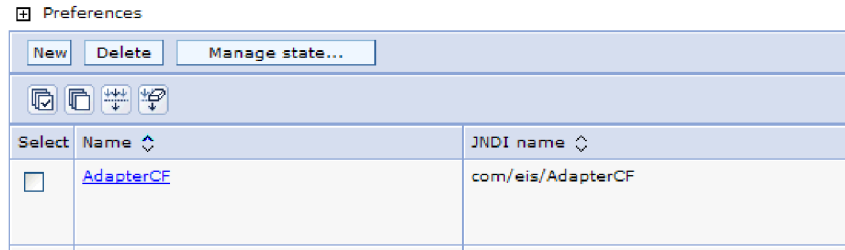
Figure 20. User-defined connection factories for use with the resource adapter

8. In the list of connection factories, click the one you want to use.

9. In the **Additional Properties** list, click **Custom properties**.

   Custom properties are those J2C connection factory properties that are unique to WebSphere Adapter for PeopleSoft Enterprise. Connection pool and advanced connection factory properties are properties you configure if you are developing your own adapter.

10. For each property you want to change, perform the following steps.

    Note: See "Managed connection factory properties" on page 127 for more information about these properties.

    a. Click the name of the property.

    b. Change the contents of the **Value** field or type a value, if the field is empty.

    c. Click **OK**.

11. After you have finished setting properties, click **Apply**.

12. In the Messages area, click **Save**.

Results

The managed connection factory properties associated with your adapter are set.

Setting activation specification properties for stand-alone adapters

To set activation specification properties for your stand-alone adapter after it has been installed on IBM Business Process Manager or WebSphere Enterprise Service Bus, use the administrative console. You select the name of the message endpoint property you want to configure, and then change or set the value.

Before you begin

Your adapter must be installed on IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

You use activation specification properties to configure the endpoint for inbound processing.

To configure properties using the administrative console, use the following procedure.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.

2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.

3. Log on to the administrative console.

4. Click **Resources** > **Resource Adapters** > **Resource adapters**.

5. In the Resource adapters page, click **IBM WebSphere Adapter for PeopleSoft Enterprise**.

6. In the **Additional Properties** list, click **J2C activation specifications**.

7. If you are going to use an existing activation specification, skip ahead to select from an existing list of activation specifications.

   Note: If you have selected **Use predefined connection properties** when you use the external service wizard to configure the adapter module, you must create an activation specification.

   If you are creating an activation specification, perform the following steps:

   a. Click **New**.

   b. In the **General Properties** section of the **Configuration** tab, type a name for the activation specification. For example, you can type AdapterAS.

   c. Type a value for **JNDI name**. For example, you can type com/eis/AdapterAS.

   d. Optional: Select an authentication alias from the **Authentication alias** list.

   e. Select a message listener type.

   f. Click **OK**.

   g. Click **Save** in the **Messages** box at the top of the page.

   The newly created activation specification is displayed.

8. In the list of activation specifications, click the one you want to use.

9. In the Additional Properties list, click **J2C activation specification custom properties**.

10. For each property you want to set, perform the following steps.

    Note: See "Activation specification properties" on page 142 for more information about these properties.

    a. Click the name of the property.

    b. Change the contents of the **Value** field or type a value, if the field is empty.

    c. Click **OK**.

11. After you have finished setting properties, click **Apply**.

12. In the Messages area, click **Save**.

Results

The activation specification properties associated with your adapter are set.

Starting the application that uses the adapter

Use the administrative console of the server to start an application that uses the adapter. By default, the application starts automatically when the server starts.

About this task

Use this procedure to start the application, whether it is using an embedded or a stand-alone adapter. For an application that uses an embedded adapter, the adapter starts when the application starts. For an application that uses a stand-alone adapter, the adapter starts when the application server starts.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Applications** > **Application Types** > **WebSphere enterprise applications**.

   Note: The administrative console is labeled "Integrated Solutions Console".
5. Select the application that you want to start. The application name is the name of the EAR file you installed, without the .EAR file extension.
6. Click **Start**.

Results

The status of the application changes to Started, and a message stating that the application has started displays at the top of the administrative console.

Stopping the application that uses the adapter

Use the administrative console of the server to stop an application that uses the adapter. By default, the application stops automatically when the server stops.

About this task

Use this procedure to stop the application, whether it is using an embedded or a stand-alone adapter. For an application with an embedded adapter, the adapter stops when the application stops. For an application that uses a stand-alone adapter, the adapter stops when the application server stops.

Procedure

1. If the server is not running, right-click your server in the **Servers** view and select **Start**.
2. When the server status changes to **Started**, right-click the server and select **Administration** > **Run administrative console**.
3. Log on to the administrative console.
4. Click **Applications** > **Application Types** > **WebSphere enterprise applications**.

   Note: The administrative console is labeled "Integrated Solutions Console".
5. Select the application that you want to stop. The application name is the name of the EAR file you installed, without the .EAR file extension.
6. Click **Stop**.

Results

The status of the application changes to Stopped, and a message stating that the application has stopped is displayed at the top of the administrative console.

Monitoring performance using Performance Monitoring Infrastructure

Performance Monitoring Infrastructure (PMI) is a feature of the administrative console that allows you to dynamically monitor the performance of components in the production environment, including IBM WebSphere Adapter for PeopleSoft Enterprise. PMI collects adapter performance data, such as average response time and total number of requests, from various components in the server and organizes the data into a tree structure. You can view the data through the Tivoli® Performance Viewer, a graphical monitoring tool that is integrated with the administrative console in IBM Business Process Manager or WebSphere Enterprise Service Bus.

About this task

You can monitor the performance of your adapter by having PMI collect data at the following points:

- At outbound processing to monitor outbound requests.
- At inbound event retrieval to monitor the retrieval of an event from the event table.
- At inbound event delivery to monitor the delivery of an event to the endpoint or endpoints.

Before you enable and configure PMI for your adapter, you must first set the level of tracing detail and run some events from which to gather performance data.

To learn more about how PMI can help you monitor and improve the overall performance of your adapter environment, search for PMI on the IBM Business Process Manager or WebSphere Enterprise Service Bus website: http://www.ibm.com/software/webservers/appserv/was/library/.

Configuring Performance Monitoring Infrastructure

You can configure Performance Monitoring Infrastructure (PMI) to gather adapter performance data, such as average response time and total number of requests. After you configure PMI for your adapter, you can monitor the adapter performance using Tivoli Performance viewer.

Before you begin

Before you can configure PMI for your adapter, you must first set the level of tracing detail and run some events to gather the performance data.

1. To enable tracing and to receive event data, the trace level must be set to either fine, finer, finest, or all. After *=info, add a colon and a string, for example:

   ```
   *=info: WBILocationMonitor.CEI.ResourceAdapter.
   *=finest: WBILocationMonitor.LOG.ResourceAdapter.*=finest:
   ```

   For instructions on setting the trace level, see "Enabling tracing with the Common Event Infrastructure" on page 98.
2. Generate at least one outbound request or inbound event to produce performance data that you can configure.

Procedure

1. Enable PMI for your adapter.

    a. In the administrative console, expand **Monitoring and Tuning**, and then select **Performance Monitoring Infrastructure (PMI)**.

    b. From the list of servers, click the name of your server.

    c. Select the Configuration tab, and then select the **Enable Performance Monitoring (PMI)** check box.

    d. Select **Custom** to selectively enable or disable statistics.
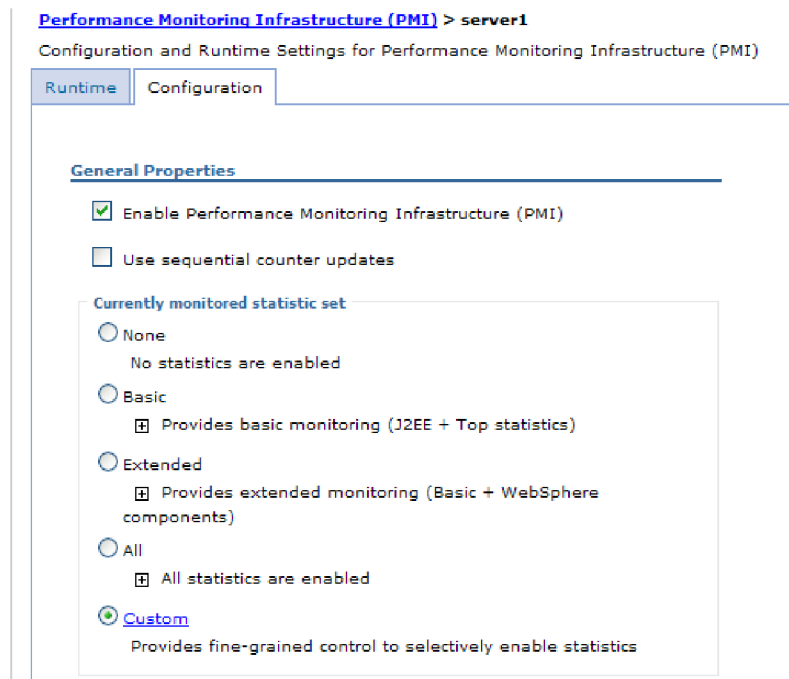
Figure 21. Enabling Performance Monitoring Infrastructure

    e. Click **Apply** or **OK**.

    f. Click **Save**. PMI is now enabled.

2. Configure PMI for your adapter.

    a. In the administrative console, expand **Monitoring and Tuning**, and then select **Performance Monitoring Infrastructure (PMI)**.

    b. From the list of servers, click the name of your server.

    c. Select **Custom**.

    d. Select the **Runtime** tab. The following figure shows the Runtime tab.
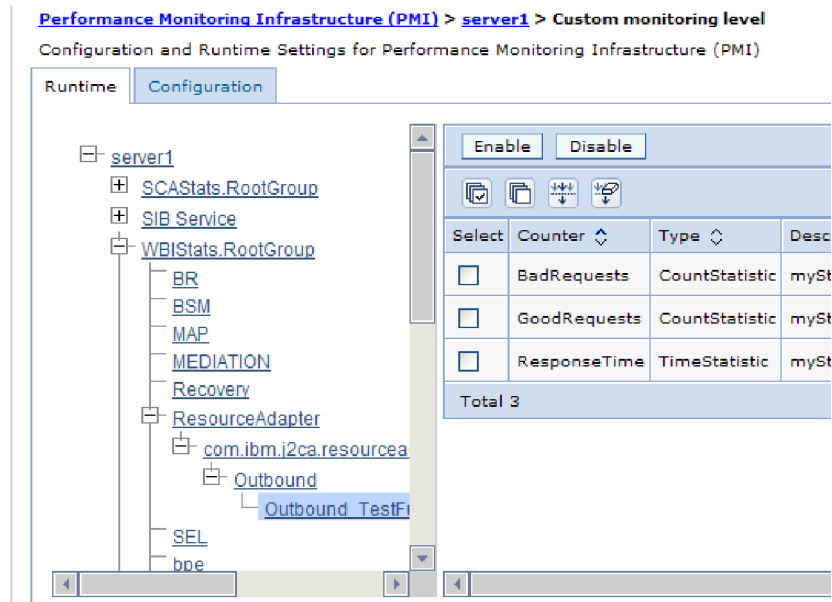
Figure 22. Runtime tab used for configuring PMI

e. Click **WBIStats.RootGroup**. This is a PMI sub module for data collected in the root group. This example uses the name WBIStats for the root group.

f. Click **ResourceAdapter**. This is a sub module for the data collected for the JCA adapters.

g. Click the name of your adapter, and select the processes you want to monitor.

h. In the right pane, select the check boxes for the statistics you want to gather, and then click **Enable**.

Results

PMI is configured for your adapter.

What to do next

Now you can view the performance statistics for your adapter.

Viewing performance statistics

You can view adapter performance data through the graphical monitoring tool, Tivoli Performance Viewer. Tivoli Performance Viewer is integrated with the administrative console in IBM Business Process Manager or WebSphere Enterprise Service Bus.

Before you begin

Configure Performance Monitoring Infrastructure for your adapter.

Procedure

1. In the administrative console, expand **Monitoring and Tuning**, expand **Performance Viewer**, then select **Current Activity**.

2. In the list of servers, click the name of your server.

3. Under your server name, expand **Performance Modules**.

4. Click **WBIStatsRootGroup**.

5. Click **ResourceAdapter** and the name of your adapter module.

6. If there is more than one process, select the check boxes for the processes whose statistics you want to view.

Results

The statistics are displayed in the right panel. You can click **View Graph** to view a graph of the data, or **View Table** to see the statistics in a table format.

The following figure shows adapter performance statistics.
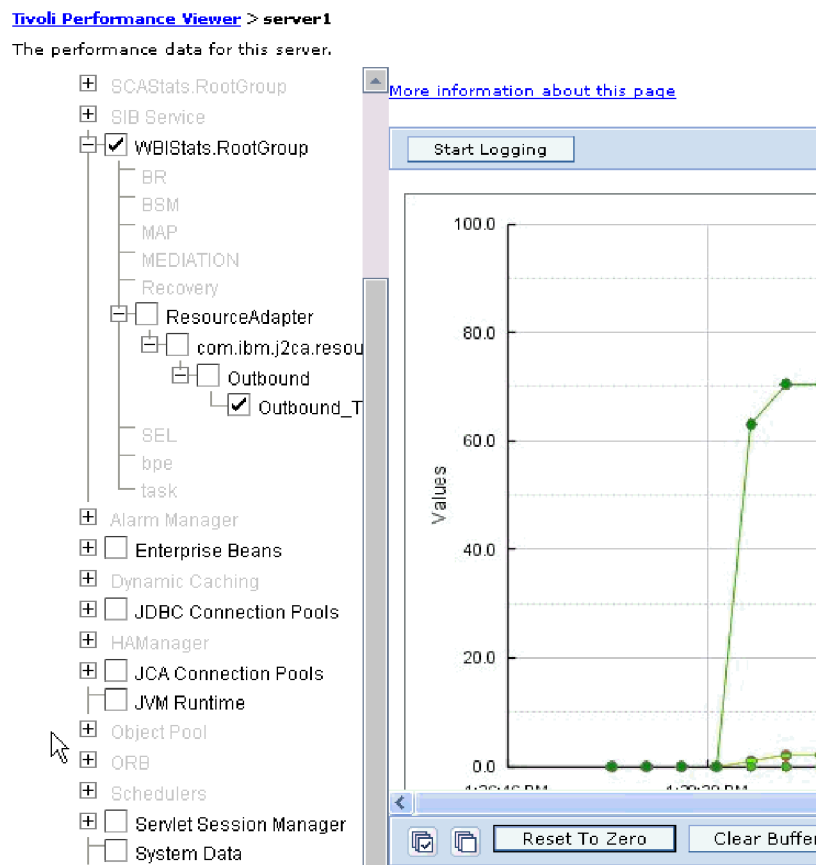
Figure 23. Adapter performance statistics, using graph view

Enabling tracing with the Common Event Infrastructure

The adapter can use the Common Event Infrastructure (CEI), a component embedded in the server, to report data about critical business events such as the starting or stopping of a poll cycle. Event data can be written to a database or a trace log file depending on configuration settings.

About this task

Use this procedure to report CEI entries in the trace log file by using the Common Base Event Browser within the administrative console.

Procedure

1. In the administrative console, click **Troubleshooting**.
2. Click **Logs and Trace**.
3. From the list of servers, click the name of your server.
4. In the **Change Log Detail Levels** box, click the name of the CEI database (for example, WBIEventMonitor.CEI.ResourceAdapter.*) or the trace log file (for example, WBIEventMonitor.LOG.ResourceAdapter.*) to which you want the adapter to write event data.
5. Select the level of detail about business events that you want the adapter to write to the database or trace log file, and (optionally) adjust the granularity of detail associated with messages and traces.
   - **No Logging**. Turns off event logging.
   - **Messages Only**. The adapter reports an event.
   - **All Messages and Traces**. The adapter reports details about an event.
   - **Message and Trace Levels**. Settings for controlling the degree of detail the adapter reports about the business object payload associated with an event. If you want to adjust the detail level, select one of the following options:

     Fine. The adapter reports the event but none of the business object payload.

     Finer. The adapter reports the event and the business object payload description.

     Finest. The adapter reports the event and the entire business object payload.
6. Click **OK**.

Results

Event logging is enabled. You can view CEI entries in the trace log file or by using the Common Base Event Browser within the administrative console.

Adding dependency libraries to the deployed resource adapter

The deployed resource adapter running in the WebSphere Application Server requires the same dependency libraries as it does in IBM Integration Designer to process requests. The method for adding these library files depends on the mode of the resource adapter deployment: standalone or embedded in the EAR file.

Stand-alone deployment

The dependency libraries may be added to the resource adapter deployed stand-alone either during initial deployment of the RAR file or by configuring the Resource Adapter properties after deployment. To set the values during initial deployment of the RAR file, specify Class path and Native path locations. Class path is used to point to JAR files, and Native path is used to point to native libraries, such as *.dll, *.so. To set the dependency library path files after the adapter has been installed on WebSphere Application Server, use the administrative console to modify the values for the Resource Adapter.

EAR deployment

For the rare case when the connector needs to be embedded in the EAR file, the dependant libraries are added as shared libraries. Define the appropriate shared library containing software dependency files and associate them with the EAR file.

About this task

There are two methods to do this task:
* Using enhanced EAR editor in IBM Integration Designer
* Using administrative console of the IBM Business Process Manager or WebSphere Enterprise Service Bus

Using enhanced EAR editor

You can use the EAR editor in IBM Integration Designer to add the dependency libraries.

About this task

To create shared libraries using the EAR editor, use the following procedure.

Procedure

1. Open Enhanced EAR editor.
2. Click the **Deployment** tab.
3. Expand **Shared Library** section.
4. Click **Add** to add new shared library.
5. Specify the shared library parameters and click **OK**.
6. Deploy the EAR to the server.

Results

The dependent libraries are added as shared libraries.

Using administrative console of the WebSphere Application Server

You can use the administrative console of the IBM Business Process Manager or WebSphere Enterprise Service Bus to add the dependency libraries.

Before you begin

Ensure that the dependent files are available on the server machine in the separate folder. If needed, copy the dependent files on the server machine.

Procedure

1. Define WebSphere variables to point to appropriate folders.
2. Define the shared library through the server administrative console; you can specify it using WebSphere variables defined in step 1.
3. Deploy the EAR to the server.
4. Configure the EAR to reference defined shared library.

Results

The dependent libraries are added as shared libraries.

Chapter 9. Troubleshooting and support

Common troubleshooting techniques and self-help information help you identify and solve problems quickly. For information about configuring logging properties and changing the log and trace file names, see "Configuring logging and tracing" on page 81.

Techniques for troubleshooting problems

Troubleshooting is a systematic approach to solving a problem. The goal is to determine why something does not work as expected and how to resolve the problem. Certain common techniques can help with the task of troubleshooting.

The first step in the troubleshooting process is to describe the problem completely. Without a problem description, neither you or IBM® can know where to start to find the cause of the problem. This step includes asking yourself basic questions, such as:
* What are the symptoms of the problem?
* Where does the problem occur?
* When does the problem occur?
* Under which conditions does the problem occur?
* Can the problem be reproduced?

The answers to these questions typically lead to a good description of the problem, and that is the best way to start down the path of problem resolution.

What are the symptoms of the problem?

When starting to describe a problem, the most obvious question is "What is the problem?" Which might seem like a straightforward question; however, you can break it down into several more-focused questions that create a more descriptive picture of the problem. These questions can include:
* Who, or what, is reporting the problem?
* What are the error codes and messages?
* How does the system fail? For example, is it a loop, hang, lock up, performance degradation, or incorrect result?
* What is the business impact of the problem?

Where does the problem occur?

Determining where the problem originates is not always simple, but it is one of the most important steps in resolving a problem. Many layers of technology can exist between the reporting and failing components. Networks, disks, and drivers are only a few components to be considered when you are investigating problems.

The following questions can help you to focus on where the problem occurs in order to isolate the problem layer.
* Is the problem specific to one platform or operating system, or is it common for multiple platforms or operating systems?
* Is the current environment and configuration supported?

Remember that if one layer reports the problem, the problem does not necessarily originate in that layer. Part of identifying where a problem originates is

understanding the environment in which it exists. Take some time to completely describe the problem environment, including the operating system and version, all corresponding software and versions, and hardware information. Confirm that you are running within an environment that is a supported configuration; many problems can be traced back to incompatible levels of software that are not intended to run together or have not been fully tested together.

When does the problem occur?

Develop a detailed timeline of events leading up to a failure, especially for those cases that are one-time occurrences. You can most simply do this by working backward: Start at the time an error was reported (as precisely as possible, even down to the millisecond), and work backward through the available logs and information. Typically, you need to look only as far as the first suspicious event that you find in a diagnostic log; however, this is not always simple to do and takes practice. Knowing when to stop looking is especially difficult when multiple layers of technology are involved, and when each has its own diagnostic information.

To develop a detailed timeline of events, answer the following questions:
- Does the problem happen only at a certain time of day or night?
- How often does the problem happen?
- What sequence of events leads up to the time that the problem is reported?
- Does the problem happen after an environment change, such as upgrading or installing software or hardware?

Responding to these types of questions can provide you with a frame of reference in which to investigate the problem.

Under which conditions does the problem occur?

Knowing what other systems and applications are running at the time that a problem occurs is an important part of troubleshooting. These and other questions about your environment can help you to identify the root cause of the problem:
- Does the problem always occur when the same task is being performed?
- Does a certain sequence of events need to occur for the problem to surface?
- Do any other applications fail at the same time?

Answering these types of questions can help you explain the environment in which the problem occurs and correlate any dependencies. Remember that just because multiple problems might have occurred around the same time, the problems are not necessarily related.

Can the problem be reproduced?

From a troubleshooting standpoint, the "ideal" problem is one that can be reproduced. Typically with problems that can be reproduced, you have a larger set of tools or procedures at your disposal to help you investigate. Consequently, problems that you can reproduce are often simpler to debug and solve. However, problems that you can reproduce can have a disadvantage: If the problem is of significant business impact, you do not want it to recur! If possible, re-create the problem in a test or development environment, which typically offers you more flexibility and control during your investigation.

Tip: Simplify the scenario to isolate the problem to a suspected component.

The following questions can help you with reproducing the problem:
- Can the problem be re-created on a test machine?
- Are multiple users or applications encountering the same type of problem?
- Can the problem be re-created by running a single command, a set of commands, a particular application, or a stand-alone application?

First-failure data capture (FFDC) support

The adapter supports first-failure data capture (FFDC), which provides persistent records of failures and significant software incidents that occur during run time in IBM Business Process Manager or WebSphere Enterprise Service Bus.

The FFDC feature runs in the background and collects events and errors that occur at run time. The feature provides a means for associating failures to one another, allowing software to link the effects of a failure to their causes, and facilitate the quick location of the root cause of a failure. The data that is captured can be used to identify exception processing that occurred during the adapter run time.

When a problem occurs, the adapter writes exception messages and context data to a log file, which is in the *install_root*/profiles/*profile*/logs/ffdc directory.

For more information about first-failure data capture (FFDC), see the IBM Business Process Manager or WebSphere Enterprise Service Bus documentation.

Resolving memory limit exception during outbound processing

A memory limit exception encountered during outbound processing can be resolved by increasing the JVM maximum heap memory size.

About this task

When concurrently processing fifty outbound requests, the adapter may produce an out of memory error. To avoid or resolve this error, increase the JVM maximum heap default value.

Procedure

1. From the administrative console, click the name of your server.
2. In the configuration tab, under Server Infrastructure, click **Java and Process Definition>Process Definition**.
3. Click **Java Virtual Machine**.
4. In the **Maximum Heap Size** field, type 1024.
5. Click **OK**.
6. Click **Save**.
7. Log out, and then restart the server.

Results

The maximum heap size default value is increased.

Support for non-english locales during inbound processing

WebSphere Adapter for PeopleSoft Enterprise cannot process an inbound operation when the system locale is non-English (such as Chinese and Hebrew).

Problem:

Even though the system locale of the workstations where PeopleSoft server and WebSphere Adapter for PeopleSoft Enterprise run are both non-English (such as Chinese and Hebrew), the returned value is in the English locale format instead of corresponding locale format. This prevents the adapter from parsing the event time stamp.

Symptom:

The adapter fails when trying to parse the event time stamp during an inbound operation. An exception message similar to the following can be found in the ffdc log files.

```
Stack Dump = java.text.ParseException: Unparseable date: "08/31/2007 10:58AM PDT"
at com.ibm.icu.text.DateFormat.parse(DateFormat.java:488)
at com.ibm.j2ca.peoplesoft.PeopleSoftEvent.(PeopleSoftEvent.java:80) at com.ibm.j2ca.peoplesoft.PeopleSoftEventStoreWith
getEvents(PeopleSoftEventStoreWithXid.java:370) at com.ibm.j2ca.extension.eventmanagement.internal.EventManager.getEvent
(EventManager.java:348) at com.ibm.j2ca.extension.eventmanagement.internal.EventManager.pollForEvents(EventManager.java
 at com.ibm.j2ca.base.internal.PollEventManagerWorker.run(PollEventManagerWorker.java:100) at com.ibm.ejs.j2c.work.WorkP
run(WorkProxy.java:418) at com.ibm.ejs.j2c.work.AsyncWorkProxy.run(AsyncWorkProxy.java:136) at com.ibm.ws.asynchbeans.J2
run(J2EEContext.java:759) at com.ibm.ws.asynchbeans.ExecutionContextImpl.go(ExecutionContextImpl.java:85) at com.ibm.ejs
AsyncWorkProxy.run(AsyncWorkProxy.java:90) at com.ibm.ws.util.ThreadPool$Worker.run(ThreadPool.java:1469)
```

Solution:

To overcome this exception, change the DateFormat property in the ActivationSpec to "MM/dd/yyyy HH:mm".

Support for setting values with PeopleTools 8.22

WebSphere Adapter for PeopleSoft Enterprise does not set some values with PeopleTools 8.22 .

Problem

WebSphere Adapter for PeopleSoft Enterprise fails when trying to set values for the EditHistoryItems and GetDummyRows properties.

Symptom

The adapter fails when trying to set values for the EditHistoryItems and GetDummyRows properties. An exception message similar to the following one appears in the trace file:

```
Caused by:
com.ibm.j2ca.peoplesoft.exceptions.EISOperationFailedException: An error has occurred
 while setting the system properties on the EIS object.
at com.ibm.j2ca.peoplesoft.PeopleSoftUtility.setSystemPropertiesOnEisObject
(PeopleSoftUtility.java:672)
at com.ibm.j2ca.peoplesoft.commands.PeopleSoftRetrieveCommand.doRetrieve
(PeopleSoftRetrieveCommand.java:96)
... 56 more
Caused by: com.ibm.j2ca.peoplesoft.exceptions.EISOperationFailedException:
The call to the method has failed and the value could not be set.
at com.ibm.j2ca.peoplesoft.PeopleSoftUtility.set(PeopleSoftUtility.java:99)
at com.ibm.j2ca.peoplesoft.PeopleSoftUtility.setSystemPropertiesOnEisObject
(PeopleSoftUtility.java:662)
... 57 more
```

```
Caused by: java.lang.NoSuchMethodException: PeopleSoft.Generated.
CompIntfc.WbiUserCompCi.setEditHistoryItems(boolean)at java.lang.Class.
throwNoSuchMethodException(Class.java:274)
at java.lang.Class.getMethod(Class.java:799)at
 com.ibm.j2ca.peoplesoft.PeopleSoftUtility.set
(PeopleSoftUtility.java:74)
```

Cause

PeopleTools 8.22 does not support the EditHistoryItems and GetDummyRows
properties.

Solution

To overcome this exception, do not use PeopleTools 8.22 for EditHistoryItems or
GetDummyRows properties.

Adapter returns version conflict exception message

Adapter returns version conflict exception message

Problem

When you install multiple adapters with different versions of
CWYBS_AdapterFoundation.jar, and if a lower version of the
CWYBS_AdapterFoundation.jar is loaded during run time, the adapter returns the
ResourceAdapterInternalException error message, due to a version conflict. For
example, when you install Oracle E-Business Suite adapter version 7.0.0.3 and
WebSphere Adapter for PeopleSoft Enterprise version 7.5.0.3, the following error
message is displayed "The version of CWYBS_AdapterFoundation.jar is not
compatible with IBM WebSphere Adapter for PeopleSoft Enterprise" as IBM
WebSphere Adapter for PeopleSoft Enterprise loads file:/C:/IBM/WebSphere/
ProcServer7/profiles/ProcSrv01/installedConnectors/CWYOE_OracleEBS.rar/
CWYBS_AdapterFoundation.jar with version 7.0.0.3. However, the base level of this
jar required is version 7.5.0.3. To overcome this conflict, you must migrate all
adapters to the same version level.

Solution

Migrate all adapters to the same version level.

For further assistance, visit http://www.ibm.com/support/
docview.wss?uid=swg27006249.

Disabling end point applications of the passive adapter

Problem

In the active-passive configuration mode of the adapters, the endpoint application
of the passive adapter instance also listens to the events or messages even if the
enableHASupport property is set to True.

Cause

By default, in WebSphere Application Server, the alwaysactivateAllMDBs property
in the JMS activation specification is set to True. This enables the endpoint
application of all the adapter (active or passive) instances to listen to the events.

Solution

To stop the endpoint application of the passive adapter instance from listening to the events, you must set the `alwaysactivateAllMDBs` property value to `False`. The JMS activation specification is associated with one or more MDBs and provides the necessary configuration to receive events. If the `alwaysActivateAllMDBs` property is set to `False`, then the endpoint application of only the active adapter instance receives the events.

Perform the following procedure, to set the `alwaysActivateAllMDBs` property to `False`.

1. Log on to the administrative console.
2. Go to **Resources> JMS > Activation specifications**.
3. Click the activation specification corresponding to the application from the list.
4. Click **Custom properties** under **Additional properties**.
5. Click **alwaysActivateAllMDBs**.
6. Change the value to **False**.
7. Click **Apply** and **OK**.

Result

The endpoint application of only the active adapter instance listens to the events.

Error displayed when using external service discovery wizard

Error displayed when using external service wizard

Problem

Error is displayed when using the external service wizard.

1. Add a PeopleSoft driver file in the Locate the Required Files and Libraries window.
2. Click **Next** and move to the Specify the Discovery Properties window.
3. Click **Back** and go to the Locate the Required Files and Libraries. Then add another PeopleSoft driver file.
4. Click **Next** and move to the Specify the Discovery Properties window.

IBM WebSphere Adapter for PeopleSoft Enterprise returns the following error: 'Failure in connection to EIS. Failed to load the driver: com.ibm.db2.jcc.DB2Driver. Check that the PeopleSoft driver classes are in the class path'.

Cause

This is an Eclipse limitation for class loading.

Solution

Close the connector project and reopen it to add the PeopleSoft driver files again.

The Add and Remove buttons in the EMD wizard do not function properly

The Add and Remove buttons in the external service wizard do not function properly

Problem

When the external service wizard is run with the same PeopleSoft RAR file consecutively, and the imported PeopleSoft driver libraries are selected to be removed, all the libraries, even those not selected, are removed. For example, the DB2® driver libraries consists of a driver JAR file and a license JAR file. During the second run, if you try to remove one of the JAR files, both the JAR files are removed.

Cause

This is caused because of an issue with IBM Integration Designer.

Unable to invoke adapter through webservices

Problem

After configuring the adapter, you might note that:
1. the webservices client based on the WSDL is not getting generated properly in IBM Integration Designer.
2. you may not be able to invoke the WSDL using certain webservices client, such as RESTUI firefox plugin and soapUI tool.

Solution

Perform the following steps to enable the adapter module to load the ASI file.
1. Create a library project.
2. Change to the Enterprise Explorer view in IBM Integration Designer.
3. Locate the ASI file of the adapter in **connector project -> connectorModule**.
4. Copy the ASI file of the adapter and paste it in the library project.
5. Add the library project to the list of dependencies for the adapter module.
6. Clean project.

Support

This section provides information about how to troubleshoot a problem with your IBM® software, including instructions for searching knowledge bases, downloading fixes, and obtaining support.

Searching knowledge bases (Web search)

You can often find solutions to problems by searching IBM knowledge bases. You can optimize your results by using available resources, support tools, and search methods.

About this task

You can find useful information by searching the information center for Product X. However, sometimes you need to look beyond the information center to answer your questions or resolve problems.

To search knowledge bases for information that you need, use one or more of the following approaches:

- Search for content by using the IBM® Support Assistant (ISA).

  ISA is a no-charge software serviceability workbench that helps you answer questions and resolve problems with IBM software products. You can find instructions for downloading and installing ISA on the ISA website.

- Find the content that you need by using the IBM Support Portal.

  The IBM Support Portal is a unified, centralized view of all technical support tools and information for all IBM systems, software, and services. The IBM Support Portal lets you access the IBM electronic support portfolio from one place. You can tailor the pages to focus on the information and resources that you need for problem prevention and faster problem resolution. Familiarize yourself with the IBM Support Portal by viewing the demo videos (https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos) about this tool. These videos introduce you to the IBM Support Portal, explore troubleshooting and other resources, and demonstrate how you can tailor the page by moving, adding, and deleting portlets.

- Search for content by using the IBM masthead search. You can use the IBM masthead search by typing your search string into the Search field at the top of any ibm.com® page.

- Search for content by using any external search engine, such as Google, Yahoo, or Bing. If you use an external search engine, your results are more likely to include information that is outside the ibm.com domain. However, sometimes you can find useful problem-solving information about IBM products in newsgroups, forums, and blogs that are not on ibm.com.

Tip: Include "IBM" and the name of the product in your search if you are looking for information about an IBM product.

Getting Fixes

A product fix might be available to resolve your problem.

About this task

To get product fixes, perform the following steps.

Procedure

1. Determine which fix you need. Check the list of IBM WebSphere Adapter for PeopleSoft Enterprise recommended fixes to confirm that your software is at the latest maintenance level. Check the list of problems fixed in the IBM WebSphere Adapter for PeopleSoft Enterprise fix readme documentation that is available for each listed fix pack to see if IBM has already published an individual fix to resolve your problem. To determine what fixes are available using IBM Support Assistant, run a query on fix from the search page.

   Individual fixes are published as often as necessary to resolve defects in IBM Business Process Manager or WebSphere Enterprise Service Bus IBM WebSphere Adapter for PeopleSoft Enterprise. In addition, two kinds of

cumulative collections of fixes, called fix packs and refresh packs, are published periodically for IBM WebSphere Adapter for PeopleSoft Enterprise, in order to bring users up to the latest maintenance level. You should install these update packages as early as possible in order to prevent problems.

Note: A list of recommended, generally available (GA) fixes for the WebSphere Java™ Connector Architecture (JCA) and WebSphere Business Integration adapters are available here. If a Fix Pack is not available for an adapter, it implies that the GA version is the recommended version and details about that version of the adapter can be found in the Release notes.

2. Download the fix. Open the download document and follow the link in the Download package section. When downloading the file, ensure the name of the maintenance file is not changed. This includes both intentional changes and inadvertent changes caused by certain web browsers or download utilities.

3. Apply the fix. Follow the instructions in the Installation Instructions section of the download document.

4. Optional: To receive weekly notification of fixes and updates, subscribe to My Support e-mail updates.

Self-help resources

Use the resources of IBM software support to get the most current support information, obtain technical documentation, download support tools and fixes, and avoid problems with WebSphere Adapters. The self-help resources also help you diagnose problems with the adapter and provide information about how to contact IBM software support.

Support website

The WebSphere Adapters software support website at http://www.ibm.com/support/docview.wss?uid=swg27006249 provides links to many resources to help you learn about, use, and troubleshoot WebSphere Adapters, including:
* Flashes (alerts about the product)
* Technical information including the product information center, manuals, IBM Redbooks®, and whitepapers
* Educational offerings
* Technotes

Recommended fixes

A list of recommended fixes you must apply is available at the following location: http://www.ibm.com/support/docview.wss?fdoc=aimadp&rs=695 &uid=swg27010397.

Technotes

Technotes provide the most current documentation about WebSphere Adapter for PeopleSoft Enterprise, including the following topics:
* Problems and their currently available solutions
* Answers to frequently asked questions
* How to information about installing, configuring, using, and troubleshooting the adapter
* *IBM Software Support Handbook*

For a list of technotes for WebSphere Adapter for PeopleSoft Enterprise, see http://www-01.ibm.com/support/docview.wss?uid=swg27024035.

For a list of technotes for all adapters, see http://www.ibm.com/support/
search.wss?tc=SSMKUK&rs=695&rank=8
&dc=DB520+D800+D900+DA900+DA800+DB560&dtm.

Chapter 10. Reference information

To support you in your tasks, reference information includes details about business objects that are generated by the external service wizard and information about adapter properties, including those that support bidirectional transformation. It also includes pointers to adapter messages and related product information.

Business object information

A business object is a structure that contains application-specific information about how the adapter process the business object and the operation to be performed on the business object. The name of the business object is generated by the external service wizard in accordance with the naming convention for the adapter.

Application-specific information

Application-specific information (ASI) is the metadata that specifies how to process business objects for WebSphere Adapter for PeopleSoft Enterprise.

When the external service wizard generates a business object, it automatically generates a business object definition, which is saved as an XSD (XML Schema Definition) file. The business object definition contains the application-specific information for that business object. If you want to change the generated ASI, you can modify the metadata values either from the Properties tab in the Business Integration perspective of IBM Integration Designer or by using the business object editor.

The adapter for PeopleSoft Enterprise uses application-specific information (ASI) to create queries for create, retrieve, update, and delete operations.

Metadata at the business-object-level

Metadata at the Business-object-level defines the top-level wrapper of the business object. The following table describes the elements of business-object metadata.

Table 3. Metadata for business objects

Metadata element	Description
ObjectName	The corresponding object in the PeopleSoft application; in other words, the component interface name as defined in PeopleTools. Set this object for the top-level business object only.
GetCurrentItem	When set to `true` at the business object level for both the parent and child business objects and the KeepRelations property is set to `true` for the element of type of child business object and EffectiveDate set to `true` for element called 'EFFDT' in child business object, only effective-dated items are retrieved during an operation.
IgnoreEffectiveSequence	When set to `true`, the effective sequence number is ignored and the record is inserted either at the lowest index, which has the current effective date or at the highest index, which has the oldest effective date. If you set this parameter, you must also set either InsAtCurrentEffDtPos or InsAtOldestEffDtPos. This parameter is used only for the effective dated items.

Table 3. Metadata for business objects (continued)

Metadata element	Description
InsAtOldestEffDtPos	When set to `true` and an incorrect index (negative number) is returned by the PeopleTools API, getEffectiveItemNum(), the adapter inserts the row at the highest index, which has the oldest effective date. If you set this parameter, do not set InsAtCurrentEffDtPos. In this case, set EffectiveDate to `true` for 'EFFDT' element in the child BO, set InteractiveMode, GetHistoryItems, and EditHistoryItems flag to `true`, set KeepRelations to `true` for child business object.
InsAtCurrentEffDtPos	When set to `true` for both parent and child business objects, EffectiveDate is set to `true` for 'EFFDT' element in the child business object, InteractiveMode, GetHistoryItems and EditHistoryItems is set to `true` and KeepRelations is set to `true` for child business object and an incorrect index (negative number) is returned by the PeopleTools API, getEffectiveItemNum(), the adapter inserts the row at the lowest index (zero), which has the current effective date. If you set this parameter, do not set InsAtOldestEffDtPos.
StatusColumnName	A property used to perform deletion of the business objects. Set this property for the top-level business object only.
StatusValue	Identifies a business object that is inactive or that is deleted. Set this property for the top-level business object only.

Metadata at the property-level

Metadata at the property level defines the property of a business object. The property can be a simple string type, or it can be a complex property type that represents child objects or an array of child objects. The following table describes the metadata elements of a complex property (a child) or a structure or table property (an array of child objects).

Table 4. Metadata for Property

Metadata element	Description
PrimaryKey	GetKey/CreateKey attribute in the component interface or a key in the child collection.
findKey	FindKey attribute in the component interface.
EffectiveDate	EffectiveDate attribute in the component interface.
EffectiveSequence	EffectiveSequence attribute in the component interface.
KeepRelations	Affects the processing of child objects when the adapter performs an update operation. If this parameter is enabled and the adapter finds a missing child object of a parent during the operation, the child object is not deleted and the relationship between the objects is preserved.
UID	Determines whether the adapter sets values for key attributes within the business object or not. When this value is set to true, the adapter does not set values for these keys but allows PeopleCode to automatically generate values for the keys.
Getter	The name of the get method used to read values for the property.
Setter	The name of the set method used to set values for the property.
FindKeyNotTopLevel	Indicates if the findKey is a level 0 property. When this property is set to true, the findkey is not in level 0 in the EIS. The default value is false which, indicates that the property is in level 0. This ASI is used by the adapter to set the EIS value during the Create or Update operation for the top-level BO. If this value is set to true, then the adapter will not set the EIS value in top-level BO. **Note:** This ASI value is set automatically by the adapter.

Supported operations

An operation is the action that an adapter can perform on the PeopleSoft
Enterprise server during outbound processing. The name of the operation typically
indicates the type of action that the adapter takes, such as *create* or *update*.

The following table defines the operations that the adapter supports.

Table 5. Supported operations of business objects

Operation	Definition
Create	The adapter accesses the PeopleSoft component and retrieves values from the attributes that have the primary key application-specific information set. It then instantiates the corresponding component interface (using the value provided for the ObjectName application-specific information). It sets the attribute values on the corresponding Create Keys in the component interface. An empty Component Interface is created, and the adapter maps all the business object data to the created component interface. When mapping the data, the adapter sends all data for simple attributes in the hierarchy, and it creates items that match each of the child objects in the hierarchy as well, including effective-dated and effective-sequenced child records.
Retrieve	The adapter accesses the PeopleSoft component and retrieves values from the attributes that have the primary key application-specific information set. It then instantiates the corresponding component interface using the value provided for the ObjectName application-specific information. It sets the attribute values on the corresponding Get Keys in the component interface. The adapter then maps the component data onto the business object hierarchy. Child objects are included in the data mapping.
RetrieveAll	This functions as the Retrieve operation, except that it allows retrieval of multiple instances of the same PeopleSoft component.
Update	The adapter retrieves an object from PeopleSoft and compares it to the target business object. It processes the objects as follows: When the comparison reveals extra child objects in PeopleSoft, the children are deleted. When the comparison reveals missing children in PeopleSoft, the children are created. When the comparison reveals child objects updated in PeopleSoft, they are updated.
Exists	The adapter processes an exist operation in the same way that it processes a retrieve operation, except that it does not populate the business object with retrieved data. It checks for the existence of an object in PeopleSoft.
Delete	Based on the values set for the application-specific metadata elements StatusColumnName and StatusValue, the adapter updates a business object to inactive. A delete operation can only be performed on a top-level object. PeopleSoft does not allow an object to be physically deleted, so the inactive object remains in the PeopleSoft database.
Apply Changes	Apply changes Updates the PeopleSoft component based on the operation that was performed on it. The supported operations are create, update, and delete.

Fault business objects listing

The adapter supports business faults, which are exceptions that are anticipated and
declared in the outbound service description, or import. Business faults occur at
predictable points in a business process, and are caused by a business rule
violation or a constraint violation.

The WebSphere Adapter for PeopleSoft Enterprise enables any faults for you.
Manual configuration of faults is not required. The adapter provides the following
fault business objects that the wizard creates:

- DuplicateRecordFault

When processing a Create operation, the adapter throws this fault if a duplicate record is found.

- MatchesExceededLimitFault

  When processing a RetrieveAll operation, the adapter returns this fault if the number of records returned from the database query exceeds the value for the MaxRecords property in the interaction specification.

  To increase the number of records that are returned, increase the value of the MaxRecords property in the interaction specification properties for the RetrieveAll operation.

  The business object for this fault has only one property, matchCount, which is a string that contains the number of matches.

- MissingDataFault

  If the business object that is passed to the outbound operation does not have all the required attributes, the adapter returns this fault. This fault can occur for the Create, Delete, Update, Retrieve, ApplyChanges and Exists operations.

- MultipleMatchingRecordsFault

  When processing a Retrieve or Update operation, the adapter returns this fault if the query returns more than one record for the keys specified. The business object for this fault has one property, matchCount, which is a string that contains the number of matches.

- RecordNotFoundFault

  When retrieving data, the adapter returns this fault if the record is not found in the database for the keys specified. This fault can occur for the Retrieve, RetrieveAll, Delete, and Update operations.

PeopleCode for a custom event project

Two PeopleCode functions are required to support inbound processing. If you create a custom event project in PeopleTools for inbound support, add the PeopleCode functions to the project.

The following PeopleCode contains the IBMPublishEvent and IBMPublishFutureDatedEvent functions that are used to publish events to the event table. Calls to these functions are made from the SavePostChange PeopleCode function in the PeopleSoft component of interest.

```
/* IBM event notification */
Component string &KEYSTRING;
Component string &KEYNAME;
Component array of string &KEYARRAY;
Component string &KEYDELIM;
Component string &IBMVERB;
Local Record &IBMREC;

Function IBMPublishFutureDatedEvent(&BO, &KEYS, &EFFDATE)
; /* == create a new record object for cw_event_tbl == */
   &IBMREC = CreateRecord(Record.IBM_EVENT_TBL);
 /* =============== KEYS ================ */
/* composing keys and values in name value format */
   &KEYSTRING = "";
   &KEYDELIM = ":";
   &KEYARRAY = Split(&KEYS, &KEYDELIM);
   &LEN = &KEYARRAY.Len;
   For &I = 1 To &LEN;
/* get keys and values */
/* get rid of record name */
      &POS1 = Find(".", &KEYARRAY [&I]);
      &L1 = Len(&KEYARRAY [&I]);
      &POS2 = &L1 - &POS1;
      &KEYNAME = Right(&KEYARRAY [&I], &POS2);
/****The code below will remove special characters and****/
/****adjust the characters' case to ensure it is same as the****/
/****attribute name in the business object definition***/
/****Start****/
      &lLen = Len(&KEYNAME);
      &sOrigString = &KEYNAME;
```

```
            &sNewString = "";
            &lCtr2 = 1;
            &isSpecialChar = "true";
            For &lCtr = 1 To &lLen;
                &sChar = Substring(&sOrigString, &lCtr, 1);
                If (&sChar = "A" Or
                        &sChar = "a" Or
                        &sChar = "B" Or
                        &sChar = "b" Or
                        &sChar = "C" Or
                        &sChar = "c" Or
                        &sChar = "D" Or
                        &sChar = "d" Or
                        &sChar = "E" Or
                        &sChar = "e" Or
                        &sChar = "F" Or
                        &sChar = "f" Or
                        &sChar = "G" Or
                        &sChar = "g" Or
                        &sChar = "H" Or
                        &sChar = "h" Or
                        &sChar = "I" Or
                        &sChar = "i" Or
                        &sChar = "J" Or
                        &sChar = "j" Or
                        &sChar = "K" Or
                        &sChar = "k" Or
                        &sChar = "L" Or
                        &sChar = "l" Or
                        &sChar = "M" Or
                        &sChar = "m" Or
                        &sChar = "N" Or
                        &sChar = "n" Or
                        &sChar = "O" Or
                        &sChar = "o" Or
                        &sChar = "P" Or
                        &sChar = "p" Or
                        &sChar = "Q" Or
                        &sChar = "q" Or
                        &sChar = "R" Or
                        &sChar = "r" Or
                        &sChar = "S" Or
                        &sChar = "s" Or
                        &sChar = "T" Or
                        &sChar = "t" Or
                        &sChar = "U" Or
                        &sChar = "u" Or
                        &sChar = "V" Or
                        &sChar = "v" Or
                        &sChar = "W" Or
                        &sChar = "w" Or
                        &sChar = "X" Or
                        &sChar = "x" Or
                        &sChar = "Y" Or
                        &sChar = "y" Or
                        &sChar = "Z" Or
                        &sChar = "z" Or
                        &sChar = "1" Or
                        &sChar = "2" Or
                        &sChar = "3" Or
                        &sChar = "4" Or
                        &sChar = "5" Or
                        &sChar = "6" Or
                        &sChar = "7" Or
                        &sChar = "8" Or
                        &sChar = "9" Or
                        &sChar = "0") Then
                    If (&isSpecialChar = "true") Then
                        &sNewString = &sNewString | Upper(&sChar);
                        &isSpecialChar = "false";
                    Else
                        &sNewString = &sNewString | Lower(&sChar);
                    End-If;
                Else
                    &isSpecialChar = "true";
                End-If;
            End-For;
            &KEYNAME = &sNewString;
/*********End*********/
            &KEYSTRING = &KEYSTRING | &KEYNAME | "=" | @&KEYARRAY [&I] | &KEYDELIM
        End-For;
        &KEYSTRING = RTrim(&KEYSTRING, ":");
        &IBMREC.IBM_OBJECT_KEYS.Value = &KEYSTRING;
/*============== VERB =======================*/
/* verb determination uses variable &IBMVERB */
        Evaluate %Mode
        When = "A"
            &IBMVERB = "Create";
            Break;
        When = "U"
            &IBMVERB = "Update";
```

```
      Break;
   When = "L"
      &IBMVERB = "Update";
      Break;
   When = "C"
      &IBMVERB = "Update";
      Break;
   When-Other
      &IBMVERB = "Retrieve";
   End-Evaluate;
   &IBMREC.IBM_OBJECT_VERB.Value = &IBMVERB;
/* ====================== EVENT_ID GEN   ================================== */
/* create event_id */
   &NEWNUM = GetNextNumber(IBM_FETCH_ID.IBM_NEXT_EVENT_ID, 99999);
/* only use newnum if no error generating next number  */
   If &NEWNUM > 0 Then
      &IBMREC.IBM_EVENT_ID.Value = &NEWNUM;
   Else
      &IBMREC.IBM_EVENT_ID.Value = %Datetime;
   End-If; /*Support for Future Effective Date - The adapter will poll such events when the date arrives*/
   If &EFFDATE > %Datetime Then
      &IBMREC.IBM_EVENT_DTTM.Value = &EFFDATE;
      &IBMREC.IBM_EVENT_STATUS.Value = "99";
   Else
      &IBMREC.IBM_EVENT_DTTM.Value = %Datetime;
      &IBMREC.IBM_EVENT_STATUS.Value = "0";
   End-If; /*================  INSERT EVENT INTO IBM_EVENT_TBL ============*/
/* insert row into table using record object*/
   &IBMREC.IBM_OBJECT_NAME.Value = &BO;
   &IBMREC.Insert();
End-Function;

Function IBMPublishEvent(&BO, &KEYS);
   /* == create a new record object for cw_event_tbl == */
   &IBMREC = CreateRecord(Record.IBM_EVENT_TBL);


   /* =============== KEYS ================ */
   /* composing keys and values in name value format */
   &KEYSTRING = "";
   &KEYDELIM = ":";
   &KEYARRAY = Split(&KEYS, &KEYDELIM);
   &LEN = &KEYARRAY.Len;

   For &I = 1 To &LEN;
      /* get keys and values */
      /* get rid of record name */
      &POS1 = Find(".", &KEYARRAY [&I]);
      &L1 = Len(&KEYARRAY [&I]);
      &POS2 = &L1 - &POS1;
      &KEYNAME = Right(&KEYARRAY [&I], &POS2);

      /****The code below will remove special characters and
      /****adjust the characters' case to ensure it is same as the
      /****attribute name in the business object definition***/
      /****Start****/
      &lLen = Len(&KEYNAME);
      &sOrigString = &KEYNAME;
      &sNewString = "";
      &lCtr2 = 1;
      &isSpecialChar = "true";
      For &lCtr = 1 To &lLen;
         &sChar = Substring(&sOrigString, &lCtr, 1);
         If (&sChar = "A" Or
                &sChar = "a" Or
                &sChar = "B" Or
                &sChar = "b" Or
                &sChar = "C" Or
                &sChar = "c" Or
                &sChar = "D" Or
                &sChar = "d" Or
                &sChar = "E" Or
                &sChar = "e" Or
                &sChar = "F" Or
```

```
                    &sChar = "f" Or
                    &sChar = "G" Or
                    &sChar = "g" Or
                    &sChar = "H" Or
                    &sChar = "h" Or
                    &sChar = "I" Or
                    &sChar = "i" Or
                    &sChar = "J" Or
                    &sChar = "j" Or
                    &sChar = "K" Or
                    &sChar = "k" Or
                    &sChar = "L" Or
                    &sChar = "l" Or
                    &sChar = "M" Or
                    &sChar = "m" Or
                    &sChar = "N" Or
                    &sChar = "n" Or
                    &sChar = "O" Or
                    &sChar = "o" Or
                    &sChar = "P" Or
                    &sChar = "p" Or
                    &sChar = "Q" Or
                    &sChar = "q" Or
                    &sChar = "R" Or
                    &sChar = "r" Or
                    &sChar = "S" Or
                    &sChar = "s" Or
                    &sChar = "T" Or
                    &sChar = "t" Or
                    &sChar = "U" Or
                    &sChar = "u" Or
                    &sChar = "V" Or
                    &sChar = "v" Or
                    &sChar = "W" Or
                    &sChar = "w" Or
                    &sChar = "X" Or
                    &sChar = "x" Or
                    &sChar = "Y" Or
                    &sChar = "y" Or
                    &sChar = "Z" Or
                    &sChar = "z" Or
                    &sChar = "1" Or
                    &sChar = "2" Or
                    &sChar = "3" Or
                    &sChar = "4" Or
                    &sChar = "5" Or
                    &sChar = "6" Or
                    &sChar = "7" Or
                    &sChar = "8" Or
                    &sChar = "9" Or
                    &sChar = "0") Then
               If (&isSpecialChar = "true") Then
                  &sNewString = &sNewString | Upper(&sChar);
                  &isSpecialChar = "false";
               Else
                  &sNewString = &sNewString | Lower(&sChar);
               End-If;
            Else
               &isSpecialChar = "true";
            End-If;
        End-For;
        &KEYNAME = &sNewString;

        /*********End*********/
```

```
                &KEYSTRING = &KEYSTRING | &KEYNAME | "=" | @&KEYARRAY [&I] | &KEYDELIM
            End-For;
            &KEYSTRING = RTrim(&KEYSTRING, ":");

            &IBMREC.IBM_OBJECT_KEYS.Value = &KEYSTRING;

            /*============== VERB =========================*/
            /*  verb determination uses variable &IBMVERB */
            Evaluate %Mode
            When = "A"
                &IBMVERB = "Create";
                Break;
            When = "U"
                &IBMVERB = "Update";
                Break;
            When = "L"
                &IBMVERB = "Update";
                Break;
            When = "C"
                &IBMVERB = "Update";
                Break;
            When-Other
                &IBMVERB = "Retrieve";
            End-Evaluate;

            &IBMREC.IBM_OBJECT_VERB.Value = &IBMVERB;

            /*  ====================  EVENT_ID GEN   ============================= */
            /*  create event_id */

            &NEWNUM = GetNextNumber(IBM_FETCH_ID.IBM_NEXT_EVENT_ID, 99999);

            /*  only use newnum if no error generating next number  */

            If &NEWNUM > 0 Then
                &IBMREC.IBM_EVENT_ID.Value = &NEWNUM;
            Else
                &IBMREC.IBM_EVENT_ID.Value = %Datetime;
            End-If;

            &IBMREC.IBM_EVENT_DTTM.Value = %Datetime;

            /* ==============  EVENT_STATUS  =================*/
            /* Validate and set event status &IBMSTATUS - list values if date is ok*/
            &IBMREC.IBM_EVENT_STATUS.Value = "0";

            /*================  INSERT EVENT INTO IBM_EVENT_TBL ============*/
            /*  insert row into table using record object*/


            &IBMREC.IBM_OBJECT_NAME.Value = &BO;

            &IBMREC.Insert();

        End-Function;
```

Outbound configuration properties

IBM WebSphere Adapter for PeopleSoft Enterprise has several categories of outbound connection configuration properties, which you set with the external service wizard while generating or creating objects and services. You can change the resource adapter and managed connection factory properties after you deploy the module to IBM Business Process Manager or WebSphere Enterprise Service Bus

using IBM Integration Designer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

Guide to information about properties

The properties used to configure WebSphere Adapter for PeopleSoft Enterprise are described in detail in tables included in each of the configuration properties topics, such as Resource adapter properties, Managed connection factory properties, and so on. To help you use these tables, information about each row you might see is explained here.

The following table explains the meaning of each row that might be displayed in the table for a configuration property.

Row	Explanation
Required	A required field (property) must have a value in order for the adapter to work. Sometimes the external service wizard provides a default value for required properties.
	Removing a default value from a required field on the external service wizard *will not change that default value*. When a required field contains no value at all, the external service wizard processes the field using its assigned default value, and that default value is displayed on the administrative console.
	Possible values are **Yes** and **No**.
	Sometimes a property is required only when another property has a specific value. When this is the case, the table notes this dependency. For example,
	• Yes, when the EventQueryType property is set to Dynamic
	• Yes, for Oracle databases
Possible values	Lists and describes the possible values that you can select for the property.
Default	The predefined value that is set by the external service wizard. When the property is required, you must either accept the default value or specify one yourself. If a property has no default value, the table states No default value.
	The word None is an acceptable default value, and does not mean that there is no default value.
Unit of measure	Specifies how the property is measured, for example in kilobytes or seconds.
Property type	Describes the property type. Valid property types include:
	• Boolean
	• String
	• Integer

Row	Explanation
Usage	Describes usage conditions or restrictions that might apply to the property. For instance, here is how a restriction would be documented: For Rational® Application Developer for WebSphere Software version 6.40 or earlier, the password: • Must be uppercase • Must be 8 characters in length For versions of Rational Application Developer for WebSphere Software later than 6.40, the password: • Is not case-sensitive • Can be up to 40 characters in length. This section lists other properties that affect this property or the properties that are affected by this property and describes the nature of the conditional relationship.
Example	Provides sample property values, for example: "If Language is set to JA (Japanese), code page number is set to 8000".
Globalized	If a property is globalized, it has national language support, meaning that you can set the value in your national language. Valid values are **Yes** and **No**.
Bidi supported	Indicates whether the property is supported in bidirectional (bidi) processing. Bidirectional processing refers to the task of processing data that contains both right-to-left (Hebrew or Arabic, for example) and left-to-right (a URL or file path, for example) semantic content within the same file. Valid values are **Yes** and **No**.

Connection properties for the external service wizard

Connection properties for the external service wizard are used to establish a connection between the external service wizard and the application from which the wizard obtains metadata. These properties specify such things as connection configuration, bidirectional (bidi) properties, and logging options.

Note: If you set any of these connection properties using bidirectional script, you must set values that identify the format of the bidirectional script entered for that property.

The external service wizard connection properties and their purpose are described in the following table. A complete description of each property is provided in the sections that follow the table.

Table 6. external service wizard connection properties

Property name	Description
"Host name " on page 121	Specifies the name or address of the server that hosts PeopleSoft Enterprise.
"Password " on page 121	Specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.
"Port number" on page 121	The port number at which PeopleSoft Enterprise is configured to listen for client requests.

Table 6. external service wizard connection properties (continued)

"Prefix for business object names" on page 122	Specifies a prefix to be added to generated business objects.
"User name" on page 122	The name of the user account the adapter uses on the PeopleSoft Enterprise server.

Host name

This property specifies the name or address of the server that hosts PeopleSoft Enterprise.

Table 7. Host name details

Required	Yes
Default	No default value
Property type	String
Usage	Identifies the server, either by name or IP address, that hosts PeopleSoft Enterprise.
Example	`9.26.248.202`
Globalized	No
Bidi supported	No

Password

This property specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.

Table 8. Password details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the PeopleSoft Enterprise version.
Globalized	Yes
Bidi supported	Yes

Port number

The port number at which PeopleSoft Enterprise is configured to listen for client requests.

Table 9. Port number details

Required	Yes
Default	The port number entered when you run the external service wizard.
Property type	Integer
Example	`9000`
Globalized	No
Bidi supported	No

Prefix for business object names

This property specifies a prefix to be added to generated business objects.

Table 10. Prefix details

Required	No
Default	No default
Property type	String
Usage	This property allows you to distinguish between different business objects generated against the same PeopleSoft component interface.
Example	If you used IB as a prefix, all business objects generated by this service are named using this prefix.
Globalized	Yes
Bidi supported	No

User name

This property specifies the name of the user account the adapter uses on the PeopleSoft Enterprise server.

Table 11. User name details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the PeopleSoft Enterprise version.
Example	DV1
Globalized	Yes
Bidi supported	Yes

Resource adapter properties

The resource adapter properties control the general operation of the adapter. You set the resource adapter properties using the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

The following properties for logging and tracing are deprecated:
- LogFileMaxSize
- LogFileName
- LogNumberOfFiles
- TraceFileMaxSize
- TraceFileName
- TraceNumberOfFiles

The following table lists the resource adapter properties and their purpose. A complete description of each property is provided in the sections that follow the table.

Table 12. Resource adapter properties for the Adapter for PeopleSoft Enterprise

Name		Description
In the wizard	**In the administrative console**	
"Adapter ID (adapterID)"	adapterID	Identifies the adapter instance for PMI events and for logging and tracing.
Disguise user data as "XXX" in log and trace files	hideConfidentialTrace	Specifies whether to disguise potentially sensitive information by writing X strings instead of user data in the log and trace files.
(Not available)	enableHASupport	Do not change this property.
(Not available)	LogFileSize	Deprecated
(Not available)	LogFilename	Deprecated
(Not available)	LogNumberOfFiles	Deprecated
(Not available)	TraceFileSize	Deprecated
(Not available)	TraceFileName	Deprecated
(Not available)	TraceNumberOfFiles	Deprecated

Adapter ID (adapterID)

This property identifies a specific deployment or instance of the adapter.

Table 13. Adapter ID details

Required	Yes
Default	001
Property type	String

Table 13. Adapter ID details (continued)

Usage	This property identifies the adapter instance in the log and trace files, and also helps identify the adapter instance while monitoring adapters. The adapter ID is used with an adapter-specific identifier, 001, to form the component name used by the Log and Trace Analyzer tool. For example, if the adapter ID property is set to 001, the component ID is PSFTRA001. If you run multiple instances of the same adapter, ensure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate the log and trace information to a particular adapter instance. By making the first seven characters of an adapter ID property unique, the component ID for multiple instances of that adapter is also unique, allowing you to correlate the log and trace information to a particular instance of an adapter. For example, when you set the adapter ID property of two instances of WebSphere Adapter for PeopleSoft Enterprise to 001 and 002. The component IDs for those instances, PSFTRA001 and PSFTRA002, are short enough to remain unique, enabling you to distinguish them as separate adapter instances. However, instances with longer adapter ID properties cannot be distinguished from each other. If you set the adapter ID properties of two instances to Instance01 and Instance02, you will not be able to examine the log and trace information for each adapter instance because the component ID for both instances is truncated to PSFTRAInstanc. For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	Yes
Bidi supported	No

Disguise user data as "XXX" in log and trace files (hideConfidentialTrace)

This property specifies whether to replace user data in log and trace files with a string of X's to prevent unauthorized disclosure of potentially sensitive data.

Table 14. Disguise user data as "XXX" in log and trace files details

Required	No
Possible values	True False
Default	False
Property type	Boolean

Table 14. Disguise user data as "XXX" in log and trace files details (continued)

Usage	If you set this property to True, the adapter replaces user data with a string of X's when writing to log and trace files.
	For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	No
Bidi supported	No

Enable high availability support (enableHASupport)

Do not change this property. It must be set to true.

Log file maximum size (LogFileMaxSize)

This property specifies the size of the log files in kilobytes.

Table 15. Log file maximum size details

Required	No
Default	0
Property type	Integer
Usage	When the log file reaches it maximum size, the adapter start using a new log file. If the file size is specified as 0 or no maximum size is specified, the file does not have a maximum size.
Globalized	Yes
Bidi supported	No

Log file name (LogFilename)

This property specifies the full path name of the log file.

Table 16. Log file name details

Required	No
Default	No default value
Property type	String
Usage	This property is deprecated.
Globalized	Yes
Bidi supported	Yes

Log number of files (LogNumberOfFiles)

This property specifies the number of log files.

Table 17. Log number of files details

Required	No
Default	1
Property type	Integer
Usage	When a log file reaches its maximum size, the adapter starts using another log file. If no value is specified, the adapter creates a single log file.
Globalized	Yes
Bidi supported	No

Trace file maximum size (TraceFileMaxSize)

This property specifies the size of the trace files in kilobytes.

Table 18. Trace file maximum size details

Required	No
Default	0
Property type	Integer
Usage	If no value is specified, then the trace file has no maximum size.
Globalized	Yes
Bidi supported	No

Trace file name (TraceFilename)

This property specifies the full path of the trace file.

Table 19. Trace file name details

Required	No
Default	No default value
Unit of measure	Kilobytes
Property type	String
Usage	This property is deprecated.
Globalized	Yes
Bidi supported	Yes

Trace number of files (TraceNumberOfFiles)

This property specifies the number of trace files to use. When a trace file reaches its maximum size, the adapter starts using another trace file.

Table 20. Trace number of files details

Required	No
Default	1
Property type	Integer

Table 20. Trace number of files details (continued)

Usage	If no value is specified, the adapter uses a single trace file.
Globalized	Yes
Bidi supported	No

Managed connection factory properties

Managed connection factory properties are used by the adapter at run time to create an outbound connection with the PeopleSoft Enterprise server.

The following table lists and describes the managed connection factory properties for outbound communication. You set the managed connection factory properties using either the external service wizard or the administrative console (after deployment).

A more detailed description of each property is provided in the sections that follow the table. For information about how to read the property details tables in the sections that follow, see "Guide to information about properties" on page 119.

Note: The external service wizard refers to these properties as managed connection factory properties and the administrative console refers to them as (J2C) connection factory properties.

Table 21. Managed connection factory properties

In the wizard	In the administrative console	Description
"Adapter ID (adapterID)" on page 128	adapterID	Identifies the adapter instance for PMI events and for logging and tracing.
Component interface for testing failed connection	PingCompInterface	Specifies the component interface the adapter uses to validate a connection to the PeopleSoft Enterprise server.
Disguise user data as "XXX" in log and trace files	hideConfidential Trace	Specifies whether to disguise potentially sensitive information by writing X strings instead of user data in the log and trace files.
"Show error on empty result set (showErrorOnEmptyResultSet)" on page 130	showErrorOnEmpty ResultSet	Specifies whether to generate a fault or return an empty business object when no records are found.
Host name	hostName	Specifies the name or address of the server that hosts PeopleSoft Enterprise.
"Language (language)" on page 130	language	Specifies the language code the adapter uses to log on to the PeopleSoft Enterprise server.
"Maximum retries in case of system connection failure (connectionRetryLimit)" on page 130	connectionRetry Limit	Specifies the maximum number of times the adapter attempts to reestablish an outbound connection to the EIS.
"Password (Password)" on page 131	password	Specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.
"Port number (Port)" on page 131	port	The port number the adapter uses to access the PeopleSoft Enterprise server.
"Retry interval if connection drops(in milliseconds) (connectionRetryInterval)" on page 132	connectionRetry Interval	Specifies the time interval between attempts to reconnect to the PeopleSoft Enterprise server if the connection fails.
"User name (UserName)" on page 132	userName	The name of the user account the adapter uses on the PeopleSoft Enterprise server.

Table 21. Managed connection factory properties (continued)

"Use Invalid Request Fault (useInvalidRequestFault)" on page 132	useInvalidRequest Fault	Specifies whether to generate the invalid request fault business object

Adapter ID (adapterID)

This property identifies a specific deployment or instance of the adapter.

Table 22. Adapter ID details

Required	Yes
Default	001
Property type	String
Usage	This property identifies the adapter instance in the log and trace files, and also helps identify the adapter instance while monitoring adapters. The adapter ID is used with an adapter-specific identifier, 001, to form the component name used by the Log and Trace Analyzer tool. For example, if the adapter ID property is set to 001, the component ID is PSFTRA001.
	If you run multiple instances of the same adapter, ensure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate the log and trace information to a particular adapter instance. By making the first seven characters of an adapter ID property unique, the component ID for multiple instances of that adapter is also unique, allowing you to correlate the log and trace information to a particular instance of an adapter.
	For example, when you set the adapter ID property of two instances of WebSphere Adapter for PeopleSoft Enterprise to 001 and 002. The component IDs for those instances, PSFTRA001 and PSFTRA002, are short enough to remain unique, enabling you to distinguish them as separate adapter instances. However, instances with longer adapter ID properties cannot be distinguished from each other. If you set the adapter ID properties of two instances to Instance01 and Instance02, you will not be able to examine the log and trace information for each adapter instance because the component ID for both instances is truncated to PSFTRAInstanc.
	For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	Yes
Bidi supported	No

Component interface for testing failed connection (PingCompInterface)

This property specifies the name of the PeopleSoft Enterprise component interface that the adapter uses to validate a connection to the PeopleSoft Enterprise server.

Table 23. Component interface for testing failed connection details

Required	Yes
Default	The name of the first component interface in the list.
Property type	String
Usage	Specify a component interface name that exists within your PeopleSoft Enterprise applications.
Example	WBI_CUSTOMER_CI
Globalized	No
Bidi supported	No

Disguise user data as "XXX" in log and trace files (hideConfidentialTrace)

This property specifies whether to replace user data in log and trace files with a string of X's to prevent unauthorized disclosure of potentially sensitive data.

Table 24. Disguise user data as "XXX" in log and trace files details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	If you set this property to True, the adapter replaces user data with a string of X's when writing to log and trace files. For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	No
Bidi supported	No

Host name (hostName)

This property specifies the name or address of the server that hosts PeopleSoft Enterprise.

Table 25. Host name details

Required	Yes
Default	No default value
Property type	String
Usage	Identifies, either by name or IP address, the server that hosts PeopleSoft Enterprise.
Example	9.26.248.202
Globalized	No
Bidi supported	No

Language (language)

This property specifies the language code the adapter uses to log on to the PeopleSoft Enterprise server.

Table 26. Language details

Required	Yes
Default	The default value for the Language property is based on the system locale.
Property type	String
Usage	Each of the supported languages is preceded by a three character language code. The language itself is presented in parentheses.
Example	If the system locale is English, the value for this property is ENG (English).
Globalized	No
Bidi supported	No

Show error on empty result set (showErrorOnEmptyResultSet)

This property specifies whether to return an empty business object when no records are found.

Table 27. Show error on empty result set details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If you set this property to False, the RetrieveAll operation returns an empty business object when no records are found. If you set this property to True, the RetrieveAll operation generates a RecordNotFoundFault or InvalidRequestFault depending on the value of useInvalidRequestFault.
Globalized	No
Bidi supported	No

Maximum retries in case of system connection failure (connectionRetryLimit)

This property specifies the number of times the adapter attempts to reestablish a connection to the PeopleSoft Enterprise server, when the adapter encounters an error related to the outbound connection.

Table 28. Maximum retries in case of system connection failure details

Required	No
Possible values	Integers equal to and greater than zero
Default	0
Property type	Integer

Table 28. Maximum retries in case of system connection failure details (continued)

Usage	When this property is set to:
	0 • The adapter does not attempt to reconnect to the PeopleSoft Enterprise server, if an error occurs during startup or while establishing a connection. • The adapter does not verify if the connection to the PeopleSoft Enterprise server is valid when there is an outbound request during run time. **>0** • The adapter attempts to reconnect to the PeopleSoft Enterprise server for the specified number of times, if an error occurs during startup or while establishing a connection. • The adapter verifies if the connection to the PeopleSoft Enterprise server is valid when there is an outbound request during run time. If the connection is not valid, it is terminated and a new connection is created to process the request. If the adapter fails to establish a connection after trying for the specified number of times, a connection error is generated. If the adapter is successful in reestablishing the connection, the outbound operation is completed.
Globalized	No
Bidi supported	No

Password (Password)

This property specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.

Table 29. Password details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the version of PeopleSoft Enterprise.
Globalized	No
Bidi supported	No

Port number (Port)

This property specifies the port number the adapter uses to access the PeopleSoft Enterprise server.

Table 30. Port number details

Required	Yes
Default	The port number entered when you use the external service wizard to discover objects and services.
Property type	Integer
Example	9000
Globalized	No

Table 30. Port number details (continued)

Bidi supported	No

Retry interval if connection drops(in milliseconds) (connectionRetryInterval)

This property specifies the time interval between attempts to reconnect to the PeopleSoft Enterprise server if the connection fails.

Table 31. Retry interval if connection failure details

Required	No
Possible values	Integers equal and greater than 0
Default	60000
Unit of measure	Milliseconds
Property type	Integer
Usage	This property is applicable only if the value of the property "Maximum retries in case of system connection failure (connectionRetryLimit)" on page 130 is set to greater than 0. When the adapter encounters an error while establishing a connection to the PeopleSoft Enterprise server, this property specifies the time interval the adapter waits between attempts to reestablish a connection.
Globalized	No
Bidi supported	No

User name (UserName)

This property specifies the name of the user account the adapter uses on the PeopleSoft Enterprise server.

Table 32. User name details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the PeopleSoft Enterprise version.
Example	DV1
Globalized	No
Bidi supported	No

Use Invalid Request Fault (useInvalidRequestFault)

This property specifies whether to generate the invalid request fault business object.

Table 33. Use Invalid Request Fault (useInvalidRequestFault)

Required	No
Default	False
Property type	Boolean

Table 33. Use Invalid Request Fault (useInvalidRequestFault) (continued)

Usage	If you set this property to True, the adapter returns an InvalidRequestFault business object if any error occurs in PeopleSoft Enterprise system server side during run time outbound processing. The error codes and error messages are contained in this business object.
Globalized	No
Bidi supported	No

Interaction specification properties

Interaction specification properties control the interaction for an operation. The external service wizard sets the interaction specification properties when you configure the adapter. Typically, you do not need to change these properties. However, some properties for outbound operations can be changed by the user. For example, you might increase the value of the interaction specification property that specifies the maximum number of records to be returned by a RetrieveAll operation, if your RetrieveAll operations do not return complete information. To change these properties after the application is deployed, use the editor in IBM Integration Designer. The properties reside in the method binding of the import.

Table 34 lists and describes the interaction specification property that you set. For information about how to read the property detail tables in the sections that follow, see "Guide to information about properties" on page 119.

Table 34. Interaction specification property for the Adapter for PeopleSoft Enterprise

Property name	Description
Maximum number of records for RetrieveAll operation	Maximum number of records to return during a RetrieveAll operation.

Maximum number of records for RetrieveAll operation (MaxRecords)

This property specifies the maximum number of records to return during a RetrieveAll operation.

Table 35. Maximum number of records for RetrieveAll operation details

Required	Yes
Default	100
Usage	If the number of hits in PeopleSoft Enterprise exceeds the value of the Maximum number of records for RetrieveAll operation property, the adapter returns an error. The adapter uses this property to help avoid out-of-memory issues.
Property type	Integer
Globalized	No
Bidi supported	No

Inbound configuration properties

WebSphere Adapter for PeopleSoft Enterprise has several categories of inbound connection configuration properties, which you set with the external service wizard while generating or creating objects and services. You can change the resource adapter and activation specification properties after you deploy the module using

IBM Integration Designer or the administrative console, but connection properties for the external service wizard cannot be changed after deployment.

Guide to information about properties

The properties used to configure WebSphere Adapter for PeopleSoft Enterprise are described in detail in tables included in each of the configuration properties topics, such as Resource adapter properties, Managed connection factory properties, and so on. To help you use these tables, information about each row you might see is explained here.

The following table explains the meaning of each row that might be displayed in the table for a configuration property.

Row	Explanation
Required	A required field (property) must have a value in order for the adapter to work. Sometimes the external service wizard provides a default value for required properties.
	Removing a default value from a required field on the external service wizard *will not change that default value*. When a required field contains no value at all, the external service wizard processes the field using its assigned default value, and that default value is displayed on the administrative console.
	Possible values are **Yes** and **No**.
	Sometimes a property is required only when another property has a specific value. When this is the case, the table notes this dependency. For example, • Yes, when the EventQueryType property is set to Dynamic • Yes, for Oracle databases
Possible values	Lists and describes the possible values that you can select for the property.
Default	The predefined value that is set by the external service wizard. When the property is required, you must either accept the default value or specify one yourself. If a property has no default value, the table states No default value.
	The word None is an acceptable default value, and does not mean that there is no default value.
Unit of measure	Specifies how the property is measured, for example in kilobytes or seconds.
Property type	Describes the property type. Valid property types include: • Boolean • String • Integer

Row	Explanation
Usage	Describes usage conditions or restrictions that might apply to the property. For instance, here is how a restriction would be documented: For Rational Application Developer for WebSphere Software version 6.40 or earlier, the password: • Must be uppercase • Must be 8 characters in length For versions of Rational Application Developer for WebSphere Software later than 6.40, the password: • Is not case-sensitive • Can be up to 40 characters in length. This section lists other properties that affect this property or the properties that are affected by this property and describes the nature of the conditional relationship.
Example	Provides sample property values, for example: "If Language is set to JA (Japanese), code page number is set to 8000".
Globalized	If a property is globalized, it has national language support, meaning that you can set the value in your national language. Valid values are **Yes** and **No**.
Bidi supported	Indicates whether the property is supported in bidirectional (bidi) processing. Bidirectional processing refers to the task of processing data that contains both right-to-left (Hebrew or Arabic, for example) and left-to-right (a URL or file path, for example) semantic content within the same file. Valid values are **Yes** and **No**.

Connection properties for the external service wizard

Connection properties for the external service wizard are used to establish a connection between the external service wizard and the application from which the wizard obtains metadata. These properties specify such things as connection configuration, bidirectional (bidi) properties, and logging options.

Note: If you set any of these connection properties using bidirectional script, you must set values that identify the format of the bidirectional script entered for that property.

The external service wizard connection properties and their purpose are described in the following table. A complete description of each property is provided in the sections that follow the table.

Table 36. external service wizard connection properties

Property name	Description
"Host name " on page 136	Specifies the name or address of the server that hosts PeopleSoft Enterprise.
"Password " on page 136	Specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.
"Port number" on page 136	The port number at which PeopleSoft Enterprise is configured to listen for client requests.

Table 36. external service wizard connection properties (continued)

"Prefix for business object names" on page 137	Specifies a prefix to be added to generated business objects.
"User name" on page 137	The name of the user account the adapter uses on the PeopleSoft Enterprise server.

Host name

This property specifies the name or address of the server that hosts PeopleSoft Enterprise.

Table 37. Host name details

Required	Yes
Default	No default value
Property type	String
Usage	Identifies the server, either by name or IP address, that hosts PeopleSoft Enterprise.
Example	9.26.248.202
Globalized	No
Bidi supported	No

Password

This property specifies the password of the user account of the adapter on the PeopleSoft Enterprise server.

Table 38. Password details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the PeopleSoft Enterprise version.
Globalized	Yes
Bidi supported	Yes

Port number

The port number at which PeopleSoft Enterprise is configured to listen for client requests.

Table 39. Port number details

Required	Yes
Default	The port number entered when you run the external service wizard.
Property type	Integer
Example	9000
Globalized	No
Bidi supported	No

Prefix for business object names

This property specifies a prefix to be added to generated business objects.

Table 40. Prefix details

Required	No
Default	No default
Property type	String
Usage	This property allows you to distinguish between different business objects generated against the same PeopleSoft component interface.
Example	If you used IB as a prefix, all business objects generated by this service are named using this prefix.
Globalized	Yes
Bidi supported	No

User name

This property specifies the name of the user account the adapter uses on the PeopleSoft Enterprise server.

Table 41. User name details

Required	Yes
Default	No default value
Property type	String
Usage	The restrictions (case, length, and character) are determined by the PeopleSoft Enterprise version.
Example	DV1
Globalized	Yes
Bidi supported	Yes

Resource adapter properties

The resource adapter properties control the general operation of the adapter. You set the resource adapter properties using the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

The following properties for logging and tracing are deprecated:
- LogFileMaxSize
- LogFileName
- LogNumberOfFiles
- TraceFileMaxSize
- TraceFileName
- TraceNumberOfFiles

The following table lists the resource adapter properties and their purpose. A complete description of each property is provided in the sections that follow the table.

Table 42. Resource adapter properties for the Adapter for PeopleSoft Enterprise

Name		Description
In the wizard	**In the administrative console**	
"Adapter ID (adapterID)" on page 123	adapterID	Identifies the adapter instance for PMI events and for logging and tracing.
Disguise user data as "XXX" in log and trace files	hideConfidentialTrace	Specifies whether to disguise potentially sensitive information by writing X strings instead of user data in the log and trace files.
(Not available)	enableHASupport	Do not change this property.
(Not available)	LogFileSize	Deprecated
(Not available)	LogFilename	Deprecated
(Not available)	LogNumberOfFiles	Deprecated
(Not available)	TraceFileSize	Deprecated
(Not available)	TraceFileName	Deprecated
(Not available)	TraceNumberOfFiles	Deprecated

Adapter ID (adapterID)

This property identifies a specific deployment or instance of the adapter.

Table 43. Adapter ID details

Required	Yes
Default	001
Property type	String

Table 43. Adapter ID details (continued)

| Usage | This property identifies the adapter instance in the log and trace files, and also helps identify the adapter instance while monitoring adapters. The adapter ID is used with an adapter-specific identifier, 001, to form the component name used by the Log and Trace Analyzer tool. For example, if the adapter ID property is set to 001, the component ID is PSFTRA001.

If you run multiple instances of the same adapter, ensure that the first seven characters of the adapter ID property are unique for each instance so that you can correlate the log and trace information to a particular adapter instance. By making the first seven characters of an adapter ID property unique, the component ID for multiple instances of that adapter is also unique, allowing you to correlate the log and trace information to a particular instance of an adapter.

For example, when you set the adapter ID property of two instances of WebSphere Adapter for PeopleSoft Enterprise to 001 and 002. The component IDs for those instances, PSFTRA001 and PSFTRA002, are short enough to remain unique, enabling you to distinguish them as separate adapter instances. However, instances with longer adapter ID properties cannot be distinguished from each other. If you set the adapter ID properties of two instances to Instance01 and Instance02, you will not be able to examine the log and trace information for each adapter instance because the component ID for both instances is truncated to PSFTRAInstanc.

For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries. |
| --- | --- |
| Globalized | Yes |
| Bidi supported | No |

Disguise user data as "XXX" in log and trace files (hideConfidentialTrace)

This property specifies whether to replace user data in log and trace files with a string of X's to prevent unauthorized disclosure of potentially sensitive data.

Table 44. Disguise user data as "XXX" in log and trace files details

Required	No
Possible values	True False
Default	False
Property type	Boolean

Table 44. Disguise user data as "XXX" in log and trace files details (continued)

Usage	If you set this property to `True`, the adapter replaces user data with a string of X's when writing to log and trace files. For inbound processing, the value of this property is set at the resource adapter level. For outbound processing, the value can be set both at the resource adapter level and the managed connection factory level. After you use the external service wizard to configure the adapter for outbound processing, you can set the resource adapter and managed connection factory properties independently. If you use the IBM Integration Designer assembly editor or the administrative console to reset these properties, ensure that you set them consistently, to prevent inconsistent marking of the log and trace entries.
Globalized	No
Bidi supported	No

Enable high availability support (enableHASupport)

Do not change this property. It must be set to `true`.

Log file maximum size (LogFileMaxSize)

This property specifies the size of the log files in kilobytes.

Table 45. Log file maximum size details

Required	No
Default	`0`
Property type	Integer
Usage	When the log file reaches it maximum size, the adapter start using a new log file. If the file size is specified as `0` or no maximum size is specified, the file does not have a maximum size.
Globalized	Yes
Bidi supported	No

Log file name (LogFilename)

This property specifies the full path name of the log file.

Table 46. Log file name details

Required	No
Default	No default value
Property type	String
Usage	This property is deprecated.
Globalized	Yes
Bidi supported	Yes

Log number of files (LogNumberOfFiles)

This property specifies the number of log files.

Table 47. Log number of files details

Required	No
Default	1
Property type	Integer
Usage	When a log file reaches its maximum size, the adapter starts using another log file. If no value is specified, the adapter creates a single log file.
Globalized	Yes
Bidi supported	No

Trace file maximum size (TraceFileMaxSize)

This property specifies the size of the trace files in kilobytes.

Table 48. Trace file maximum size details

Required	No
Default	0
Property type	Integer
Usage	If no value is specified, then the trace file has no maximum size.
Globalized	Yes
Bidi supported	No

Trace file name (TraceFilename)

This property specifies the full path of the trace file.

Table 49. Trace file name details

Required	No
Default	No default value
Unit of measure	Kilobytes
Property type	String
Usage	This property is deprecated.
Globalized	Yes
Bidi supported	Yes

Trace number of files (TraceNumberOfFiles)

This property specifies the number of trace files to use. When a trace file reaches its maximum size, the adapter starts using another trace file.

Table 50. Trace number of files details

Required	No
Default	1
Property type	Integer

Table 50. Trace number of files details (continued)

Usage	If no value is specified, the adapter uses a single trace file.
Globalized	Yes
Bidi supported	No

Activation specification properties

Activation specification properties hold the inbound event processing configuration information for an export. You set activation specification properties either through the external service wizard or the administrative console.

The following table lists the activation specification properties for inbound communication. A more detailed description of each property is provided in the sections that follow the table. For information about how to read the property detail tables in the sections that follow, see "Guide to information about properties" on page 119.

Table 51. Activation specification properties

Descriptive name	Property name	Purpose
Adapter Instance for event filtering	AdapterInstanceEvent Filter	Identifier that determines whether this adapter instance processes specific events in the event store.
Ensure once-only event delivery	AssuredOnceDelivery	Specifies whether the adapter provides assured once delivery of events.
Do not process events that have a timestamp in the future	FilterFutureEvents	Specifies whether the adapter filters out future events by comparing the timestamp on each event with the system time.
Component interface for testing failed connection	PingCompIntfc	Specifies the component interface the adapter uses to validate a connection to the PeopleSoft Enterprise server
Component interface name for event notification	EventCIName	Specifies the component interface the adapter uses for event notification
Delivery type	DeliveryType	Determines the order in which events are delivered by the adapter to the export.
Delimiter for keys in the event store	EventKeyDelimiter	Specifies the name and value for an object key in the event table
Event types to process	EventTypeFilter	A delimited list of event types that indicates to the adapter which events it should deliver. The adapter supports event filtering based on operations, in addition to the BO Name.
Java date format for event timestamp	DateFormat	Specifies the format used to create the event timestamp
Maximum connections	MaximumConnections	The maximum number of connections that the adapter can use for inbound event delivery.
Minimum connections	MinimumConnections	The minimum number of connections that the adapter can use for inbound event delivery.
Maximum number of retries in case of system connection failure	RetryLimit	The number of times the adapter tries to reestablish an inbound connection after an error.

Table 51. Activation specification properties (continued)

Interval between polling periods	PollPeriod	The length of time that the adapter waits between polling periods.
Poll quantity	PollQuantity	The number of events the adapter delivers to the export during each poll period.
Retry EIS connection on startup	RetryConnectionOnStartup	Controls whether the adapter retries the connection to the PeopleSoft Enterprise server if it cannot connect at startup.
Time between retries in case of system connection failure (milliseconds)	RetryInterval	The length of time that the adapter waits between attempts to reestablish connection after an error during inbound operations.
Stop the adapter when an error is encountered while polling	StopPollingOnError	Specifies whether the adapter stops polling for events when it encounters an error during polling.

Adapter Instance for event filtering (AdapterInstanceEvent Filter)

This property controls whether the adapter instance processes specific events in the event store.

Table 52. Adapter Instance for event filtering details

Required	No
Default	`null`
Property type	String

Table 52. Adapter Instance for event filtering details (continued)

Usage	This property helps you migrate from WebSphere Business Integration Adapter for PeopleSoft to WebSphere Adapter for PeopleSoft Enterprise. WebSphere Business Integration Adapter for PeopleSoft allows you to perform load balancing on high-volume event types by allowing multiple adapter instances to process events of the same type. When load balancing is not required, a single adapter instance processes all events of a given type. This property is to enable seamless migration for WBIA customers to JCA for customers who are currently taking advantage of the connectorID filtering. WebSphere Adapter for PeopleSoft Enterprise typically does not require load balancing in this way, but supports it so that you can migrate without modifying the database triggers or other mechanisms that write events to the event store. The AdapterInstanceEvent Filter property corresponds to the ConnectorID property of the WebSphere Business Integration Adapter for PeopleSoft. To use this feature, the PeopleCode that create events in the event store must assign the appropriate value to the IBM_CONNECTOR_ID column. Table 53 shows the interaction between the AdapterInstanceEvent Filter property and the value in the IBM_CONNECTOR_ID column in the event store. If the EventTypeFilter and AdapterInstanceEvent Filter properties are both set, the adapter processes only events that meet both criteria. That is, it processes only those events whose type is specified in the EventTypeFilter property and whose IBM_CONNECTOR_ID column matches the AdapterInstanceEvent Filter property.
Example	See Table 53.
Globalized	Yes
Bidi supported	Yes

Table 53. Interaction of the AdapterInstanceEvent
Filter property with the IBM_CONNECTOR_ID column in the event store

AdapterInstanceEvent Filter property	IBM_CONNECTOR_ID column of an event	Result
null	null	The adapter processes the event.
null	Instance1	The adapter processes the event, because the IBM_CONNECTOR_ID column is not checked.
Instance1	Instance1	The adapter processes the event.
Instance1	Instance2	The adapter does not process the event, because the instance IDs do not match.
Instance1	null	The adapter does not process the event, because the instance IDs do not match.

Ensure once-only event delivery (AssuredOnceDelivery)

This property specifies whether to provide ensure once-only event delivery for inbound events.

Table 54. Ensure once-only event delivery details

Required	Yes
Possible values	`True` `False`
Default	`True`
Property type	Boolean
Usage	When this property is set to `True`, the adapter provides assured once event delivery. This means that each event is delivered once and only once. A value of `False` does not provide assured once event delivery, but provides better performance. When this property is set to `True`, the adapter attempts to store transaction (XID) information in the event store. If it is set to `False`, the adapter does not attempt to store the information. This property is used only if the export component is transactional. If it is not, no transaction can be used, regardless of the value of this property.
Globalized	No
Bidi supported	No

Component interface for testing failed connection (PingCompInterface)

This property specifies the name of the PeopleSoft Enterprise component interface that the adapter uses to validate a connection to the PeopleSoft Enterprise server.

Table 55. Component interface for testing failed connection details

Row	Explanation
Required	Yes
Default	The name of the first component interface in the list.
Property type	String
Usage	The name of the component interface that the adapter uses to test connectivity to the PeopleSoft Enterprise server. Specify a component interface name that exists within your PeopleSoft Enterprise applications.
Globalized	No
Bidi supported	No

Component interface name for event notification (EventCIName)

This property specifies the name of the PeopleSoft Enterprise component interface that the adapter uses for inbound processing.

Table 56. Component interface name for event notification details

Row	Explanation
Required	Yes
Default	`IBM_EVENT_CI`
Property type	String
Usage	The name of the component interface that the adapter uses for inbound processing. To use inbound processing, you must create a component interface specifically for event notification within PeopleSoft Enterprise.

Table 56. Component interface name for event notification details (continued)

Row	Explanation
Globalized	No
Bidi supported	No

Delivery type (DeliveryType)

This property specifies the order in which events are delivered by the adapter to the export.

Table 57. Delivery type details

Required	No
Possible values	ORDERED UNORDERED
Default	ORDERED
Property type	String
Usage	The following values are supported: • ORDERED: The adapter delivers events to the export one at a time. • UNORDERED: The adapter delivers all events to the export at once.
Globalized	No
Bidi supported	No

Delimiter for keys in the event store (EventKeyDelimiter)

This property specifies the delimiter for the object key name-value pair in the event table.

Table 58. Delimiter for keys in the event store details

Row	Explanation
Required	No
Default	=:
Property type	String
Usage	This property allows you to specify an object name and value to be used as an object key in the event store.
Example	CustomerID=2001
Globalized	No
Bidi supported	No

Do not process events that have a timestamp in the future (FilterFutureEvents)

This property specifies whether the adapter filters out future events by comparing the timestamp on each event with the system time.

Table 59. Do not process events that have a timestamp in the future details

Required	Yes

Possible values	True False
Default	False
Property type	Boolean
Usage	If set to True, the adapter compares the time of each event to the system time. If the event time is later than the system time, the event is not delivered. If set to False, the adapter delivers all events.
Globalized	No
Bidi supported	No

Event types to process (EventTypeFilter)

This property contains a delimited list of event types that indicates to the adapter which events it should deliver.

Table 60. Event types to process details

Required	No
Possible values	A comma-delimited (,) list of business object types and operations.
Default	null
Property type	String
Usage	Events are filtered by business object type and operations. If the property is set, the adapter delivers only those events that are in the list. A value of null or * indicates that no filter will be applied and that all events will be delivered to the export. When the default operation set (Create/Update/Delete) is not preferred, the adapter will provide an operation-based event polling capability. For example: If you select Create|Update operation, the adapter will poll only those events with Create or Update operation for all the BOs. The default string generated for eventTypeFilter is *: Create|Update. **Syntax:** BOName:Operation1|Operation2, BOName:Operation1|Operation2|Operation3 • "," is used for separating the business objects. The objects that are not in this list are ignored. • ":" is used for separating the business object name and the operation name. • "|" is used for separating the supported operations, such as Create, Delete, and Update.

Table 60. Event types to process details (continued)

Example	• To receive events related to the Customer and Order business objects regardless of operations, specify this value: `Customer,Order` or specify this value for all supported operations: `Customer:Create|Update|Delete,Order:Create|Update|Delete`. • To receive all events of the Customer business object and the Create and Delete events of the Order business object, specify this value: `Customer,Order:Create|Delete` • To receive the Create and Delete events of all the business objects, specify this value: `*:Create|Delete`. **Note:** In this property, you can use only those operations that are chosen while selecting business objects and services. If the EventTypeFilter and AdapterInstanceEvent Filter properties are both set, the adapter processes only events that meet both criteria. That is, it processes only those events whose type is specified in the EventTypeFilter property and whose IBM_CONNECTOR_ID column matches the AdapterInstanceEvent Filter property.
Globalized	No
Bidi supported	No

Retry limit for failed events (FailedEventRetryLimit)

This property specifies the number of times that the adapter attempts to redeliver an event before marking the event as failed.

Table 61. Retry limit for failed events details

Required	No
Possible values	Integers
Default	5
Property type	Integer
Usage	Use this property to control how many times the adapter tries to send an event before marking it as failed. It accepts the following values: **Default** If this property is not set, the adapter tries five additional times before marking the event as failed. **0** The adapter tries to deliver the event an infinite number of times. When the property is set to 0, the event remains in the event store and the event is never marked as failed. **> 0** For integers greater than zero, the adapter retries the specified number of times before marking the event as failed. **< 0** For negative integers, the adapter does not retry failed events.
Globalized	No
Bidi supported	No

Java date format for event timestamp (DateFormat)

This property specifies the format that is used for the event timestamp.

Table 62. Java date format for event timestamp details

Row	Explanation
Required	Yes
Default	MM/dd/yy
Property type	String
Usage	This property is used to format the date values obtained from the PeopleSoft Enterprise server.
Globalized	No
Bidi supported	No

Maximum connections (MaximumConnections)

This property specifies the maximum number of connections that the adapter can use for inbound event delivery.

Table 63. Maximum connections details

Required	No
Default	1
Property type	Integer
Usage	Only positive values are valid. Any value less than 1 is treated as 1 by the adapter.
Globalized	No
Bidi supported	No

Minimum connections (MinimumConnections)

This property specifies the minimum number of connections that the adapter can use for inbound event delivery.

Table 64. Minimum connections details

Required	No
Default	1
Property type	Integer
Usage	Only positive values are valid. Any value less than 1 is treated as 1 by the adapter.
Globalized	No
Bidi supported	No

Interval between polling periods (PollPeriod)

This property specifies the length of time that the adapter waits between polling periods.

Table 65. Interval between polling periods details

Required	Yes

Table 65. Interval between polling periods details (continued)

Possible values	Integers greater than or equal to 0.
Default	2000
Unit of measure	Milliseconds
Property type	Integer
Usage	The poll period is established at a fixed rate, which means that if running the poll cycle is delayed for any reason (for example, if a prior poll cycle takes longer than expected to complete) the next poll cycle occurs immediately to make up for the lost time caused by the delay.
Globalized	No
Bidi supported	No

Maximum events in polling period (PollQuantity)

This property specifies the number of events that the adapter delivers to the export during each poll period.

Table 66. Maximum events in polling period details

Required	Yes
Default	10
Property type	Integer
Usage	The value must be greater than 0. If this value is increased, more events are processed per polling period and the adapter may perform less efficiently. If this value is decreased, fewer events are processed per polling period and the adapter's performance might improve slightly.
Globalized	No
Bidi supported	No

Maximum number of retries in case of system connection failure (RetryLimit)

This property specifies the number of times the adapter tries to reestablish an inbound connection.

Table 67. Maximum number of retries in case of system connection failure

Required	No
Possible values	0 and positive integers
Default	0
Property type	Integer
Usage	This property controls how many times the adapter retries the connection if the adapter cannot connect to the PeopleSoft Enterprise server to perform inbound processing. A value of 0 indicates an infinite number of retries. To control whether the adapter retries if it cannot connect to the PeopleSoft Enterprise server when it is first started, use the RetryConnectionOnStartup property.
Globalized	No
Bidi supported	No

Retry EIS connection on startup (RetryConnectionOnStartup)

This property controls whether the adapter attempts to connect again to the PeopleSoft Enterprise server if it cannot connect at startup.

Table 68. Retry EIS connection on startup details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	This property indicates whether the adapter should retry the connection to the PeopleSoft Enterprise server if the connection cannot be made when the adapter is started: • Set the property to False when you want immediate feedback about whether the adapter can establish a connection to the PeopleSoft Enterprise server, for example, when you are building and testing the application that receives events from the adapter. If the adapter cannot connect, the adapter writes log and trace information and stops. The administrative console shows the application status as Stopped. After you resolve the connection problem, start the adapter manually. • Set the property to True if you do not need immediate feedback about the connection. If the adapter cannot connect during startup, it writes log and trace information, and then attempts to reconnect, using the RetryInterval property to determine how frequently to retry and the value of the RetryLimit property to retry multiple times until that value is reached. The administrative console shows the application status as Started.
Globalized	No
Bidi supported	No

Time between retries in case of system connection failure (RetryInterval)

When the adapter encounters an error related to the inbound connection, this property specifies the length of time the adapter waits before trying to reestablish a connection.

Table 69. Retry interval details

Required	Yes
Default	2000
Unit of measure	Milliseconds
Property type	Integer
Usage	Only positive values are valid. When the adapter encounters an error related to the inbound connection, this property specifies the length of time the adapter waits before trying to establish a new connection.
Globalized	No
Bidi supported	No

Stop the adapter when an error is encountered while polling (StopPollingOnError)

This property specifies whether the adapter will stop polling for events when it encounters an error during polling.

Table 70. Stop the adapter when an error is encountered while polling details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	If this property is set to True, the adapter stops polling when it encounters an error. If this property is set to False, the adapter logs an exception when it encounters an error during polling and continues polling.
Globalized	No
Bidi supported	No

Globalization

WebSphere Adapter for PeopleSoft Enterprise is a globalized application that can be used in multiple linguistic and cultural environments. Based on character set support and the locale of the host server, the adapter delivers message text in the appropriate language. The adapter supports bidirectional script data transformation between integration components.

Globalization and bidirectional data 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 script data transformation, which refers to the task of processing data that contains both right-to-left (for example, Hebrew or Arabic) and left-to-right (for example, a URL or file path) semantic content within the same file.

Globalization

Globalized software applications are designed and developed for use within multiple linguistic and cultural environments rather than a single environment. WebSphere Adapters, IBM Integration Designer, IBM Business Process Manager, and WebSphere Enterprise Service Bus are written in Java. 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). Therefore, when data is transferred between these 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 script data 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 data, standards are used to display and process it. Bidirectional script data transformation applies only to string type data. IBM Business Process Manager and WebSphere Enterprise Service Bus use use the Windows standard format, but applications or file systems that exchange data with the server might use a different format. The adapter transforms bidirectional script data passed between the two systems so that it is accurately processed and displayed on both sides of a transaction. It transforms the script data by using a set of properties that defines the format of script data and properties that identify content or metadata to which transformation applies.

Bidirectional script data formats

IBM Business Process Manager and WebSphere Enterprise Service Bus use 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 before introducing the data to IBM Business Process Manager or WebSphere Enterprise Service Bus.

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

Table 71. Bidirectional format attributes

Letter position	Purpose	Values	Description	Default setting
1	Order schema	I	Implicit (Logical)	I
		V	Visual	
2	Direction	L	Left-to-Right	L
		R	Right-to-Left	
		C	Contextual Left-to-Right	
		D	Contextual Right-to-Left	
3	Symmetric Swapping	Y	Symmetric swapping is on	Y
		N	Symmetric swapping is off	
4	Text Shaping	S	Text is shaped	N
		N	Text is not shaped (Nominal)	
		I	Initial shaping	
		M	Middle shaping	
		F	Final shaping	
		B	Isolated shaping	
5	Numeric Shaping	H	National (Hindi)	N
		C	Contextual shaping	
		N	Numbers are not shaped (Nominal)	

Bidirectional properties that identify data for transformation

To identify business data subject to transformation, set the BiDiContextEIS property. Do this by specifying values for each of the five bidirectional format attributes (listed in the preceding table) for the property. The BiDiContextEIS property can be set for the managed connection factory and the activation specification.

Properties enabled for bidirectional data transformation

Bidirectional data transformation properties enforce the correct format of bidirectional script data exchanged between an application and integration tools and runtime environments. Once these properties are set, bidirectional script data is correctly processed and displayed in IBM Integration Designer and IBM Business Process Manager or WebSphere Enterprise Service Bus.

Connection properties for the external service wizard

The following managed connection properties control bidirectional script data transformation.

- Password
- Username

Resource adapter properties

The following activation specification properties are enabled for bidirectional script data transformation:

- Log file name
- Trace file name

Adapter messages

View the messages issued by WebSphere Adapter for PeopleSoft Enterprise at the following location.

Link to messages: http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5m1/topic/com.ibm.wbpm.ref.doc/adapt_messages.html

The displayed Web page shows a list of message prefixes. Click a message prefix to see all the messages with that prefix:

- Messages with the prefix CWYES are issued by WebSphere Adapter for PeopleSoft Enterprise
- Messages with the prefix CWYBS are issued by the adapter foundation classes, which are used by all the adapters

Related information

The following information centers, IBM Redbooks, and web pages contain related information for WebSphere Adapter for PeopleSoft Enterprise.

Information resources

- The WebSphere Business Process Management information resources web page includes links to articles, Redbooks, documentation, and educational offerings to help you learn about WebSphere Adapters: http://www-01.ibm.com/software/integration/integration-designer/.

- The WebSphere Adapters library page includes links to all versions of the documentation: http://www.ibm.com/software/integration/wbiadapters/library/infocenter/.

Information about related products

- IBM Business Process Manager, Version 7.5.1, information center, which includes IBM Business Process Manager, IBM WebSphere Enterprise Service Bus, and IBM Integration Designer information: http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5m1/index.jsp?.
- IBM Business Process Manager, Version 7.5.0, information center, which includes IBM Business Process Manager, IBM WebSphere Enterprise Service Bus, and IBM Integration Designer information: http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp.

developerWorks® resources

- WebSphere Adapter Toolkit
- WebSphere business integration zone

Video samples

To help you use WebSphere Adapters, sample videos are available to detail some of the scenarios.

You can find several WebSphere Adapters video demonstrations at this FTP location ftp://ftp.software.ibm.com/software/websphere/integration/wsa/library/videos/.

To view these video samples:

1. Download the zip file of the required sample from the FTP location.
2. Extract the downloaded zip file to your system.
3. From the extracted folder, double-click the HTML file.
4. The video sample opens in your default Web browser.

   Note: The video has a built-in audio narration. If you do not want to listen to the audio, you can lower its tab to the minimum, and slide the tab to an audio level, whenever you want the audio back.

The following sample videos are currently available at this location: ftp://ftp.software.ibm.com/software/websphere/integration/wsa/library/videos/

- AFC: Scheduling Calendar based polling
- AFC: Overcoming an AFC version conflict situation
- PeopleSoft: Handling a RecordNotFound fault behavior
- PeopleSoft: Handling an event filtering

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
Department 2Z4A/SOM1
294 Route 100
Somers, NY 10589-0100
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: (c) (your company name) (year). Portions of

this code are derived from IBM Corp. Sample Programs. (c) 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, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A complete and current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

IBM ®

Printed in USA