

IBM WebSphere Adapters
Version 7 Release 5 Fix Pack 1 (7.5.0.1)

*IBM WebSphere Adapter for Siebel
Business Applications User Guide
Version 7 Release 5 Fix Pack 1
(7.5.0.1)*



IBM WebSphere Adapters
Version 7 Release 5 Fix Pack 1 (7.5.0.1)

*IBM WebSphere Adapter for Siebel
Business Applications User Guide
Version 7 Release 5 Fix Pack 1
(7.5.0.1)*



Note

Before using this information and the product it supports, read the information in "Notices" on page 183.

November 2011

This edition applies to version 7, Release 5, Fix Pack 1 (7.5.0.1) of IBM WebSphere Adapter for Siebel Business Applications and to all subsequent releases and modifications until otherwise indicated in new editions.

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

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

© Copyright IBM Corporation 2006, 2011.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Overview of WebSphere Adapter for Siebel Business Applications 1

Hardware and software requirements	2
Technical overview	2
Outbound processing	3
Inbound processing	8
Business objects	11
The external service wizard	12
Log and Trace Analyzer	14
Business faults	15

Chapter 2. Planning for adapter implementation 17

Before you begin	17
Security	17
Support for protecting sensitive user data in log and trace files	17
User authentication	18
Deployment options	20
WebSphere Adapters in clustered environments	24
Migrating to version 7.5 of WebSphere Adapter for Siebel Business Applications	25
Migration considerations	25
Performing the migration	27
Upgrading but not migrating a project	29
Migrating WebSphere Business Integration applications	30
Migrating applications from WebSphere InterChange Server	31
Migration considerations for WebSphere Business Integration adapters	32
Migrating application artifacts from WebSphere InterChange Server	32
Migrating adapter-specific artifacts	33
Changes to the import, export, and WSDL files after migration	35

Chapter 3. Samples and tutorials 37

Chapter 4. Configuring the module for deployment. 39

Road map for configuring the module	39
Configuring the Siebel application to work with the adapter	41
Creating an authentication alias	43
Creating the project	44
Adding external software dependencies	45
Setting connection properties for the external service wizard	47
Configuring the module for outbound processing	48
Selecting business objects	49
Selecting business services	52
Configuring the selected objects	54

Setting deployment properties and generating the service	56
Configuring the module for inbound processing	62
Selecting business objects	62
Selecting business services	65
Configuring the selected objects	67
Setting deployment properties and generating the service	69

Chapter 5. Modifying artifacts 79

Modifying service import	79
Modifying service export	80

Chapter 6. Changing interaction specification properties 83

Chapter 7. Deploying the module 85

Deployment environments	85
Deploying the module for testing	85
Adding external dependencies	85
Generating and wiring a target component for testing inbound processing	87
Adding the module to the server	88
Testing the module for outbound processing using the test client	88
Deploying the module for production	89
Adding external software dependencies on the server	89
Installing the RAR file (for modules using stand-alone adapters only)	90
Exporting the module as an EAR file	91
Installing the EAR file	92

Chapter 8. Administering the adapter module 95

Changing configuration properties for embedded adapters	95
Setting resource adapter properties for embedded adapters	95
Setting managed (J2C) connection factory properties for embedded adapters	97
Setting activation specification properties for embedded adapters	99
Changing configuration properties for stand-alone adapters	101
Setting resource adapter properties for stand-alone adapters	101
Setting managed (J2C) connection factory properties for stand-alone adapters	102
Setting activation specification properties for stand-alone adapters	103
Starting the application that uses the adapter	105
Stopping the application that uses the adapter	105

Monitoring performance using Performance Monitoring Infrastructure	106	Naming conventions	128
Configuring Performance Monitoring Infrastructure	107	Fault business objects.	131
Viewing performance statistics	108	Outbound configuration properties	131
Enabling tracing with the Common Event Infrastructure	110	Connection properties for the wizard	133
Adding dependency libraries to the deployed resource adapter	110	Resource adapter properties	140
Stand-alone deployment.	110	Managed connection factory properties.	145
EAR deployment	111	Interaction specification properties	153
Chapter 9. Troubleshooting and support	113	Inbound configuration properties.	154
Configuring logging and tracing	113	Connection properties for the wizard	156
Configuring logging properties	113	Resource adapter properties	163
Changing the log and trace file names	115	Activation specification properties	168
First-failure data capture (FFDC) support	116	Globalization	177
Solutions to some common problems	116	Globalization and bidirectional data transformation	178
Self-help resources.	123	Properties enabled for bidirectional data transformation	179
Chapter 10. Reference information	125	Adapter messages.	180
Business object information.	125	Related information	180
Application-specific information	125	Video samples	181
Supported operations.	127	Notices	183
		Programming interface information	185
		Trademarks and service marks	185
		Index	187

Chapter 1. Overview of WebSphere Adapter for Siebel Business Applications

With WebSphere® Adapter for Siebel Business Applications, you can create integrated processes that exchange information with a Siebel Business Applications, without special coding.

With the adapter, an application can send requests to the Siebel Business Applications server or receive notifications of the changes from the server and also can get application data based on the notifications.

The adapter creates a standard interface to the applications and data on the Siebel Business Applications server. The application need not understand the implementation of the Siebel Business Applications or the data structures on the server. An application, for example, can send a request to the server, to query or update an Account record, represented by a Siebel business component instance. It can also receive events from the server, for example, a notification about the updates to a customer record. This provides you with improved business workflow and processes to help manage your customer relations.

The adapter complies with the Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA). JCA standardizes the way application components, application servers, and Siebel applications, such as Siebel Business Applications server, interact with each other. The adapter makes it possible for JCA-compliant application servers to connect and interact with the Siebel Business Applications server in a standard way by using Siebel Java Data Bean.

The following example assumes that you are setting up an adapter using IBM® Integration Designer and deploying the application that includes the adapter to IBM Business Process Manager or WebSphere Enterprise Service Bus.

Suppose your company uses Siebel Business Applications to coordinate its business operations (for example, customer, and order management). Siebel Business Applications includes a business function that returns a list of customers in response to a range of customer IDs. An application component might be able to use this function as part of an overall business process. For example, when the promotions department within the company sends advertising material to customers, and as part of that process, must first obtain a list of customers.

The Siebel Business Applications function does not have a web service interface. Therefore, the application component used by the promotions department must understand the low-level API and data structures of the Siebel Business Applications function to call the function. Information technology resources must create the links between the component and the Siebel Business Applications function.

With the adapter, however, you can automatically generate an interface to the Siebel Business Applications function to hide the lower-level details of the function. Depending on how you want to use the adapter, you can embed it with the deployed application or install it as a stand-alone component, to be used by more than one application. The embedded adapter is part of an application that is deployed to WebSphere Application Server. The application component interacts with the adapter instead of with the Siebel Business Applications function.

The adapter configuration, which you generate with the external service wizard within IBM Integration Designer, uses a standard interface and standard data objects. The adapter takes the standard data object sent by the application component and calls the Siebel Business Applications function. The adapter then returns a standard data object to the application component. The application component need not interact directly with the Siebel Business Applications function; it is the adapter that calls the function and returns the results.

For example, the application component that needs the list of customers sends a standard business object with the range of customer IDs to the adapter. In return, the application component receives the results (the list of customers) in the form of a standard business object. The application component need not know how the function worked or how the data was structured. The adapter performs all the interactions with the actual Siebel Business Applications function.

Similarly, the client application might want to know about a change to the data on the Siebel Business Applications server (for example, a change to a particular customer). You can generate an adapter component that polls for such events on the Siebel Business Applications server and notify client applications of the update. In this case, the interaction begins at the Siebel Business Applications server.

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-947.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_Adapters_Family.
- 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

WebSphere Adapter for Siebel Business Applications supports the exchange of information between your applications and Siebel Business Applications. The adapter supports Siebel entities, including business objects, business components, and business services. This enables you to create business processes that exchange data.

The adapter supports outbound processing (requests for data or services from an application to the Siebel Business Applications), and inbound processing (event notification from a Siebel Business Applications server to an application).

With the adapter, you can use the existing or newly created applications that run in a supported run time environment to send requests for data, and services to the Siebel Business Applications.

You can also add event-generation triggers to Siebel business objects to have notifications of events, such as the creation, update, and deletion of a record, sent to one or more of your applications.

The adapter is metadata driven. Even after you import the adapter into IBM Integration Designer, you can continue to add new business objects to the existing module. The adapter can process the new business objects without additional coding or customization.

Outbound processing

WebSphere Adapter for Siebel Business Applications supports synchronous outbound processing. Assume, the component sends a request in the form of a WebSphere business object hierarchy to the adapter. The adapter processes the request and returns a WebSphere business object hierarchy that represents the result of the operation.

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

1. The adapter extracts the metadata from the WebSphere business object hierarchy.
2. It identifies the appropriate Siebel objects to access (for example, Siebel business objects and business components, or Siebel business services, integration objects, and integration components) depending on the objects against which the artifacts were generated.
3. The adapter extracts the outbound operation to be performed from the WebSphere business object hierarchy.
4. After accessing the required Siebel objects, the adapter retrieves, updates, deletes, or creates a Siebel business component hierarchy or performs the corresponding business service method on the integration component hierarchy.
5. If there are updates (Create, Update, and/or Delete), the adapter populates that Siebel object (business or integration component hierarchy) with data from the hierarchy of WebSphere business objects.

Supported outbound operations

The adapter supports the following outbound operations: ApplyChanges, Create, Delete, Exists, Retrieve, Retrieve All, and Update.

The following table lists and describes each of these operations.

Table 1. Supported outbound operations

Operation	Description
ApplyChanges	Updates the business component based on the delta information.
Create	Creates the business component.

Table 1. Supported outbound operations (continued)

Operation	Description
Delete	Deletes the business component and its child components.
Exists	Checks for the existence of incoming business objects. The output business object, "ExistsResult" is returned with the Boolean value populated.
Retrieve	Specifies the value of the business component.
RetrieveAll	Retrieves multiple instances of the same business component. Populates the multiple instances of the same business component as the container business graph.
Update	Updates the Siebel application with the incoming business object.

Outbound operations

Application components must invoke operations such as data retrieval from a database. An operation is the action that an adapter can perform on the database during the outbound processing. The name of the operation indicates the type of action that the adapter performs, such as Create or Delete.

WebSphere Adapter for Siebel Business Applications supports the following operations during outbound processing.

ApplyChanges operation:

The ApplyChanges operation updates the business object according to the operation that was performed on it (Create, Delete, or Update). ApplyChanges provides both delta and after-image support for modifying (creating, updating, or deleting) a business object. An after-image business object contains the complete state of the business object after all the required changes have been made to it. A delta business object contains only the key values and the data to be changed. Delta business objects are used only in the operations that update the business objects.

The ApplyChanges operation is available only with business graphs. If you set the verb property of the business graph to the name of an operation, such as Create, Update, Delete, or Updatewithdelete, the adapter performs after-image processing for the ApplyChanges operation. For example, if the verb is set to Create, the adapter processes the ApplyChanges operation in the same way it processes the Create operation.

If you do not set the verb in the business graph, the adapter uses the ChangeSummary object in the business graph to update the business object. For example, a business graph has a business object and a ChangeSummary object. The business object is used for all operations, such as ApplyChanges, Create, Delete, Update, Retrieve, RetrieveAll, and Exists. The ChangeSummary object is only used for the ApplyChanges operation. The ChangeSummary object stores the operation that is performed on it and stores the value for the corresponding business object that exists in the business object hierarchy.

In this mode, the ApplyChanges operation differs from the Update operation in the following ways:

- No retrieve operation is performed before updating.
- No comparisons are made between the incoming business object and the business object in Siebel Business Applications.
- When no verb is set on the top-level business graph, the business objects in the hierarchy (including those at the top-level) are processed based on the operation that was set in the ChangeSummary object for each of the objects.
- If an operation is not set on the business object, a NoOperation command is created and the business object is processed. For example, if an operation is not set on the business object, the business component instance is retrieved from Siebel application but is not updated. The retrieval ensures that any further updates to the child business object can be processed at a later time.

When the child business objects are processed, an appropriate action is performed based on each operation of the child business object. Operations that are allowed on a child business object in ApplyChanges are Create, Delete, and Update. The following actions occur when one of these operations is found in the child business object:

- If a Create operation is found in the child business object, then the child business object is created in the Siebel application based on the relationship (for example, Picklist or multivaluelink).
- If a Delete operation is found in the child business object, then that child business object is deleted.
- If an Update operation is found in the child business object, then the child business object is updated in the Siebel application.

When the child business objects are processed, the attribute values are set at the ChangeSummary level for each business object.

- With a Create operation, the ChangeSummary object does not have any values set against its attributes, as it is used to store only old values. The ChangeSummary object stores the operation to be performed on it and any old values (existing values from the Siebel application). As it is a Create operation and the values are new, the values are set on the business object attributes.
- With an Update operation, the old values (existing values from the Siebel application) are set against the ChangeSummary attributes, and the new values (that are updated in the Siebel application) are stored against the business object attributes.
- With a Delete operation, old values (existing values from the Siebel application) are set against the ChangeSummary attributes. New values are not stored against the business object attributes, because no new values are created for the Delete operation.

Create operation:

The Create operation creates the corresponding Siebel business component hierarchy in the Siebel application.

During the Create operation, the following occurs:

- Each of the child object is created first, and then the parent object is created. After all the child objects are created, the parent object is saved. By saving the parent object after creating all the child object, you can roll back all the child objects that were created, if the top-level Create operation fails.

- Field values are set for the simple attributes.
- Each of the business component records is saved.
- In case of a failure, the adapter generates a `SiebelCreateFailedException` error.
- If the Siebel application generates a duplicate record error, the adapter generates a `DuplicateRecordException` error.
- The output business object is populated with values for the key attributes.

Each of the child container attributes is processed as follows:

- The key attributes are not set on the parent business component. The Siebel application generates the key attributes when a record is created. If the key attributes are set, Siebel returns an exception and sets the read-only fields populated with values.
- If the child container is a business component of a `PickList`, and you want to select an existing child instance, do not create a child business object. Instead, set the value on the parent container. For example, if you pick an existing `Business Address` instance under the `Account` object, set the `PrimaryAddressId` instead of creating a `Business Address` instance. However, to create a child instance, you must populate the non-key attributes in the child business component so that the adapter tries to create the corresponding child instance in the Siebel application.
- If there are static `Picklists`, then the corresponding simple attribute in the business object is required. If a value is not listed in the `Picklists` list of values, the adapter tries to populate a value. If the list is unbounded and `Insert` is enabled, the operation is successful. If the adapter is unable to populate the value by itself, the operation fails.
- With multi value links, the child record is created. If the child `Create` operation fails, the child record is rolled back. If the parent `Create` operation fails, all the child and the parent records are rolled back.
- The adapter attempts to select an existing child business component instance (for example, `Association`) based on the value of all the attributes in the corresponding child business object. If a matching child business component instance is found, it is associated with the parent. If a matching child business component instance is not found, a new child business component is created and associated with the parent business component.
- If the child `Create` operation fails, the child component is rolled back. If the parent `Create` operation fails, only the parent component is rolled back. Because association components can also be individual components, as well as child component of other components, a child component, or parent component cannot be rolled back after an associated child component has been created.

Delete operation:

The `Delete` operation deletes the top-level business object and its child objects.

The `Delete` operation results in a physical delete, and the record is removed from the Siebel application. If the key attributes (for example, `ID`) are not set correctly, the `Delete` operation fails.

Exists operation:

The `Exists` operation looks for a specified business component instance in the Siebel application.

If the specified business component instance exists, the component receives a successful response (ExistsResults) in the form of a business object. A Boolean property set in the business object confirms that the business component exists in the database. For example, if the business component instance exists in the Siebel application, the Boolean property is set to True. If the specified record does not exist, the Boolean property is set to False. If there are additional errors during processing, then a BusinessProcessingFailedException error is returned.

Retrieve operation:

The Retrieve operation retrieves the Siebel business component that corresponds to the specified key in the incoming business object.

During the Retrieve operation, the following occurs:

- The keys are set on the top-level business component to search the records. If the record is not found, an error message is logged, and the Retrieve operation fails with an exception. If the record is found, the simple attribute values are retrieved from the top-level business component and set on the business object.

Note: A simple PickList attribute is retrieved differently from other simple attributes. A simple PickList attribute is retrieved by obtaining the value of the PickListKey from the child picklist component. To retrieve a child picklist component, a FieldName value must be used. Then, the adapter uses the Restrict application-specific information to set more conditions on the child picklist component before retrieving the PickListKey value.

- The child components are retrieved based on the type of relationship they have to the parent business object in the business object hierarchy.
- The child PickList instances are retrieved. The adapter uses the From and To application-specific information to obtain the child component. Before the adapter retrieves the child component, the From attribute value from the parent component must be set against the To attribute in the child component. If the From attribute is not set, the adapter retrieves the child component without setting any values on the To attribute. For example, only executeQuery is called.
- With multi-value link and association, if the child component record is found, the attribute value is set on the parent component.

RetrieveAll operation:

The adapter uses the RetrieveAll operation to retrieve multiple instances of the same business object.

The adapter supports the RetrieveAll request for business objects that contains the key and non-key attributes. All the key and non-key attributes populated in the incoming business object determine the selection criteria for the retrieval. For business objects on which non-key attributes are set, the selection criteria is select "*". If the values of the attributes are part of the search criteria containing "*", the RetrieveAll operation is processed for that attribute. Any additional search criteria is applied to the operation if it has already been set. The business object that is returned is a container business object of the business graph.

The steps involved in processing the RetrieveAll operation are similar to the Retrieve operation. However, the adapter does not check if all the keys are set on the incoming business object.

Note: The adapter supports attributes values that contain these special characters: '(', ')', '"', "'", " ", "<", ">". The number of records that the adapter can return cannot exceed the MaxRecords property value of the WBIInteractionSpec instance. If the number of records retrieved from the Siebel application exceeds the MaxRecords value, the adapter generates the MatchesExceededLimitException error. If the records are not retrieved, the adapter generates the RecordNotFoundException error.

Update operation:

The adapter uses the Update operation to compare the business component hierarchy retrieved from Siebel with the incoming top-level business object hierarchy.

The adapter compares the object retrieved from the Siebel application with the incoming top-level business object. Depending on the results of the comparison, the creation, update, or deletion of any child object will be processed and updated. If the child Create operation fails because the record exists, the adapter returns a RecordAlreadyExistsException error.

If the Update operation fails, the adapter returns a SiebelUpdateFailedException error.

Inbound processing

WebSphere Adapter for Siebel Business Applications supports asynchronous inbound processing. The adapter polls the Siebel applications 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 component.

When the adapter detects an event for Siebel event business components or integration components, it processes it by retrieving the updated data and converting it into a business object. Then, the adapter sends the business object to the event business component. For example, if an event business component (an account) is updated, an event trigger adds an event record to the event business component. The adapter then polls the event business component, retrieves the event record, and processes it.

When the adapter finds an event for a Siebel event business component, it is processed as follows:

1. The adapter retrieves the event information from the Siebel event business component.
2. It retrieves the corresponding event business component instance hierarchy.
3. It populates the associated WebSphere business object or business graph (if it was generated) with the values it retrieves from the event business component.
4. It sends a notification to each registered application.

Note: Before inbound processing can occur, a Siebel event business component must be created in the Siebel application (IBM2 for Siebel version 7.x and IBM_EVENT for Siebel version 8) and its name specified against the corresponding property in the adapter activation specification.

Event store

The event store is a persistent cache where event records are saved until the polling adapter can process them. The adapter uses the event store to track the inbound events.

The creation, update, or deletion of an event record in Siebel Business Applications is an 'event'. Each time a business object is created, updated, or deleted, the adapter updates the status of the event in an event store. For recovery purposes, the status of each event is continuously updated by the adapter until the events are delivered to a configured export in IBM Business Process Manager or WebSphere Enterprise Service Bus.

For example, if you have a customer component and a new customer has been added to it, this signals an update. If the adapter is configured to receive the events about the new update, there are triggers attached to the Siebel end and connected to the customer component. The triggers add a record to the event business component. The record contains information about the new customer, such as the customer ID. This information is stored in the object key. The object key is the unique identifier that provides the key name and value of the event business component that is updated (for example, Id=1-20RT). The object name is the WebSphere business object name that represents the customer component (for example, AccountBG or Account). The adapter retrieves this event and the new customer information that relates to it. It then processes the event and delivers it to the export.

Each event store created by Siebel Business Applications is associated with a particular module; however the same event store can be used by multiple modules. If you want multiple modules to use the same event store, you must set the EventTypeFilter property to a different event type for each module using the event store. Set the value for the property in the ActivationSpecification instance for each module.

During inbound processing, the adapter polls the event business components from the event store at regular intervals. Each time it polls, a number of events are processed by the adapter. Events are processed in ascending order of priority and ascending order of the event time stamp. In each poll cycle, new events are picked up. The adapter retrieves the value set in the object key field for the event and loads the business object that corresponds to it. The business object, or optionally the business graph, is created from the retrieved information and is delivered to the exports.

If you set the activation specification property AssuredOnceDelivery to true, a transaction ID (XID) value is set for each event in the event store. After the event is retrieved for processing, the XID value for it is updated in the event store and displayed in the XID column in the event business component. The event is then delivered to its corresponding export, and the status is updated to show that the event has been successfully delivered. If the application is stopped or the event is not processed completely, the XID column is filled with a value. This ensures that the event is reprocessed and sent to the export. After the connection is reestablished or the adapter starts again, the adapter checks for events in the event store that have a value in the XID column. The adapter processes these events first, and then polls the other events during the poll cycles.

The adapter can either process all events or process events filtered by business object type. You set the filter through the activation specification property, EventTypeFilter. This property contains a comma-delimited list of business object types. Only the types specified in the property are processed. If the EventTypeFilter property is not set, then all the events are processed. If the FilterFutureEvents property is set to true, the adapter filters events based on the

time stamp. The adapter compares the system time in each poll cycle to the time stamp on each event. If an event is set to occur in the future, it is not processed until that time.

After an event is successfully posted and delivered to the export, the entry is deleted from the event store. Failed events (posting and delivery to the export is unsuccessful), remain in the event store and are marked -1. This prevents duplicate processing.

Event store structure for Siebel business objects and business components

The event business component stores information about the event. The information stored is used by the resource adapter during event subscription to build the corresponding business object and send it to the registered exports. The information that is stored, as well as the structure of the event store used by the adapter, is shown in the following table.

Table 2. Event store structure for Siebel event business objects and business components

Field	Description	Example
Description	Any comment associated with the event.	Account Create Event
Event ID	The ID of the event row.	Automatically generated unique ID in Siebel (for example: 1-XYZ)
Event timestamp	The time stamp for the event. The format is in <i>mm/dd/yyyy hh:mm:ss</i> .	02/24/2007 11:37:56
Event type	The type of event.	Create, Update, or Delete
Object key	A unique identifier of the business object row for which the event was created. It is a name value pair consisting of the name of the property (key name) and value.	ID=1-20RT
Object name	The name of the business object or business graph for which the event was detected.	IOAccountPRMANIICAccount
Priority	The event priority.	1
Status	The event status. This is initially set to the value for a new event and updated by the adapter as it processes the event. The status can have one of the following values: <ul style="list-style-type: none"> • 0: Identifies a new event. • 1: Identifies an event that has been delivered to an export. • -1: An error occurred while processing the event. The status cannot be null.	0
XID	The transaction ID. This is to ensure 'assured once delivery'.	None
CONNECTOR ID	The ID set for the adapter.	Siebel001

Event store structure for Siebel business services

The event is retrieved from the IBM2 event business component and the information is used to retrieve the event business component. This creates a business graph which is published to the registered exports.

Table 3. Event store structure for IBM2 Siebel business services

Field	Description	Example
Description	Any comment associated with the event.	Account PRM ANI Event
Event ID	The ID of the event row.	Automatically generated unique ID in Siebel (for example: 1-XYZ)
Event timestamp	The time stamp for the event. The format is in <i>mm/dd/yyyy hh:mm:ss</i> .	02/24/2007 11:37:56
Event type	The type of event.	Create, Update, or Delete
Object key	A unique identifier of the business object row for which the event was created. It is a name value pair consisting of the name of the property (key name) and value.	Name=TestName; Location=BGM, where 'Name' and 'Location' are the keys in the integration component. 'TestName' and 'BGM' are the values specified, and; is the event key delimiter.
Object name	The name of the business object or business graph for which the event was detected.	IOAccountPRMANIICAccount
Priority	The event priority.	1
Status	The event status. This is initially set to the value for a new event and updated by the adapter as it processes the event. The status can have one of the following values: <ul style="list-style-type: none"> • 0: Identifies a new event. • 1: Identifies an event that has been delivered to an export. • -1: An error occurred while processing the event. The status cannot be null.	0
XID	The transaction ID. This is to ensure 'assured once delivery'.	None
CONNECTOR ID	The ID set for the adapter.	Siebel001

Related tasks

“Configuring the Siebel application to work with the adapter” on page 41
Configure the Siebel application to work with the adapter by creating an event table and a Siebel business object.

Business objects

To send data or obtain data from Siebel Business Applications, the adapter 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, an employee record, or an unstructured text.

How business objects are created

You create business objects by using the external service wizard, launched from IBM Integration Designer. The external service 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 Siebel business component, and collections that occur within the top-level object are child objects of it. Information about the processed object is stored in the application-specific information for the object and each of its 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.

Related reference

“Business object information” on page 125

A business object is a structure that contains application-specific information (metadata) about how the adapter must process the business object as well as 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.

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 Siebel application 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 Siebel application server to browse the metadata repository on the 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 binding.

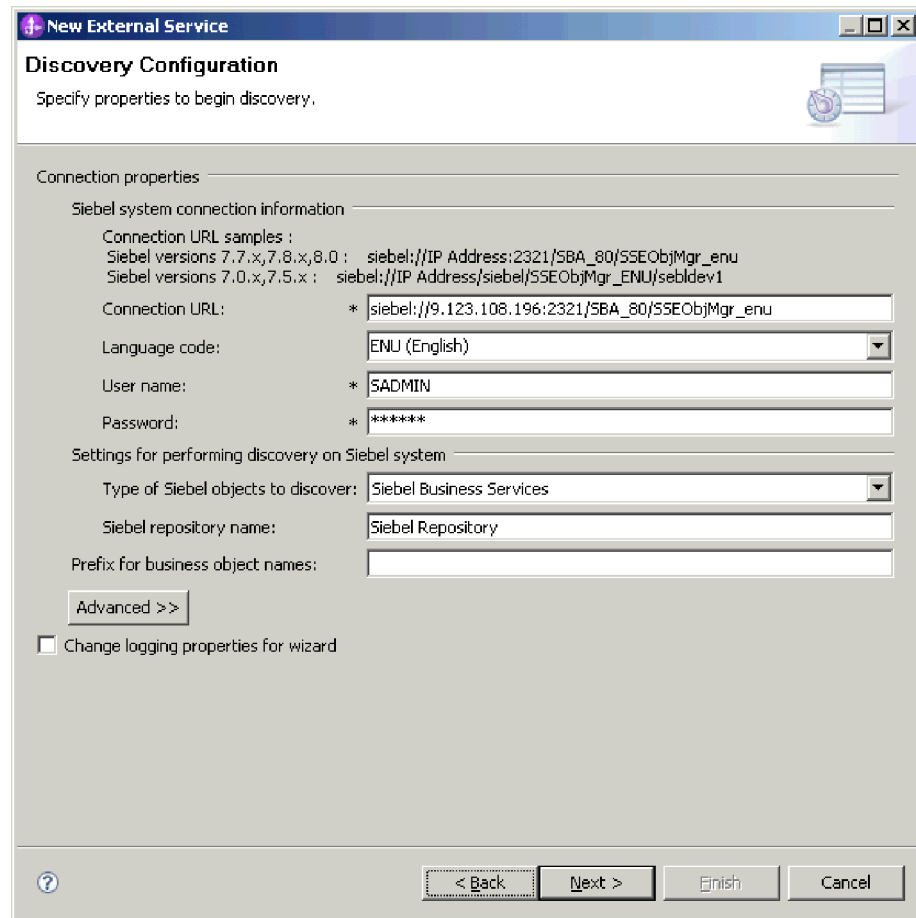


Figure 1. Configure Settings for Discovery Configuration window

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

For example, if you run the external service wizard to create a module called Outbound, you can see under Data Types, a list of generated business objects, including the objects associated with any faults that might be generated during processing.

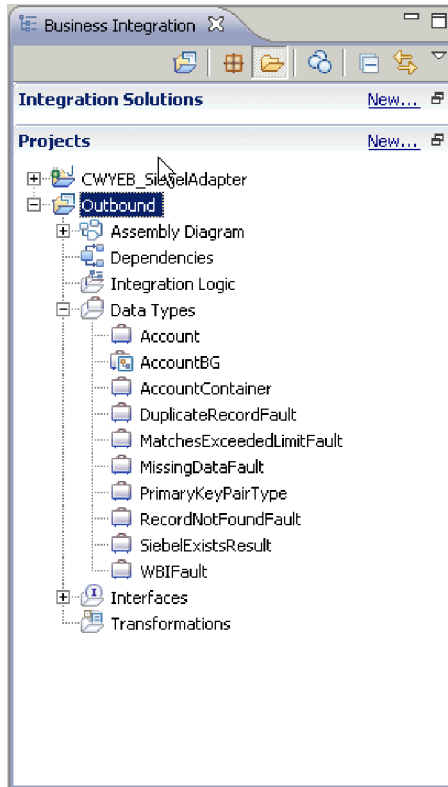


Figure 2. An example of the Outbound module generated by the external service wizard

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 adapter's messages and trace information in the log viewer.

The adapter's component ID for filtering and highlighting is a string composed of the characters SEBLRA plus the value of the adapter ID property. For example, if the adapter ID property is set to 001, the component ID is SEBLRA001.

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 Siebel Business Applications to 001 and 002. The component IDs for those instances, SEBLRA001 and SEBLRA002, 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 SEBLRAInstanc.

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 141 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 Figure 3.

WBIFault	
message	string
errorCode	string
primaryKeySet	PrimaryKeyPairType []

Figure 3. The structure of the WBIFault business object

Some faults contain the matchCount attribute that provides additional information about the error generated. For others faults, the WBIFault contains information that is required to handle the fault.

Chapter 2. Planning for adapter implementation

Before you configure WebSphere Adapter for Siebel Business Applications Software, consider whether you must set up the adapters in a clustered environment, in which the workload of the server is distributed across multiple machines. Also, if you are migrating from an earlier version of WebSphere Adapter for Siebel Business Applications Software, perform any migration tasks.

Before you begin

Before you begin to set up and use the adapter, you must possess a thorough understanding of business integration concepts, the capabilities, and requirements of the integration development tools and runtime environment you use, and the Siebel Business Applications server environment where you build and use the solution.

To configure and use WebSphere Adapter for Siebel Business Applications, you must understand and have experience with the following concepts, tools, and tasks:

- The business requirements of the solution you are building.
- The Siebel Business Applications environment in which you are working. This includes a detailed understanding of the Siebel version that supports your applications.
- Business integration concepts and models, including the Service Component Architecture (SCA) programming model.
- The capabilities provided by the integration development tools you use to build the solution. You must know how to use these tools to create modules, test components, and complete other integration tasks.
- The capabilities and requirements of the runtime environment you use for the integration solution. You must 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.
- The capabilities provided by the Siebel Business Applications version that supports your applications. You must be familiar with Siebel business objects, business components, business services, integration objects, and integration components.

Security

The adapter uses the J2C authentication data entry, or the authentication alias feature of Java 2 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 Siebel application, 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 145 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 Siebel application

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

Related reference

“Resource adapter properties” on page 140

The resource adapter properties control the general operation of the adapter. You set the resource adapter properties in the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

“Managed connection factory properties” on page 145

Managed connection factory properties are used by the adapter at run time to create an outbound connection instance with the Siebel application.

User authentication

The adapter supports several methods for supplying the user name and password that are needed to connect to the Siebel application. 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 Siebel application 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 Siebel application 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 Siebel application, although the schema name must be the same in both databases. For example, while developing and integrating an application that uses WebSphere Adapter for Siebel Business Applications, you might not use the production database; using a test database with the same data format but fewer, simulated records lets you 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 Siebel application. 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
- Data source
- 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 Siebel application. This includes the adapters embedded in application EAR files as well as adapters that are separately installed on the server.

Using a data source lets you use a connection already established for another application. For example, if multiple applications access the same database with the same user name and password, the applications can be deployed using the same data source. The user name and password can be known only to the first person who deploys an application to that data source or who defines a data source separately.

Using a J2C authentication data entry, or authentication alias, created with the Java Authentication and Authorization Service (JAAS) feature of Java 2 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.

Related tasks

“Creating an authentication alias” on page 43

The adapter can use an authentication alias to connect to the Siebel application, instead of using a user ID and password stored in an adapter property.

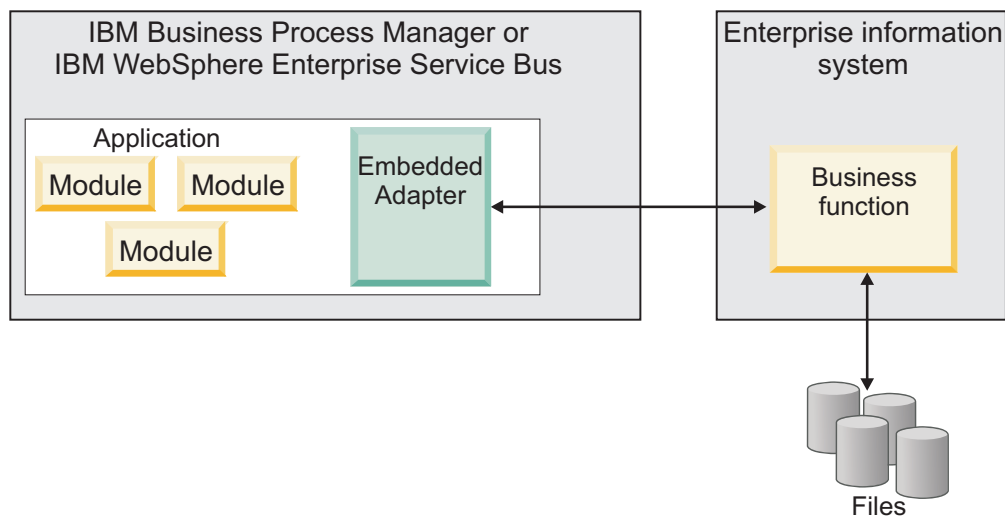
Deployment options

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.

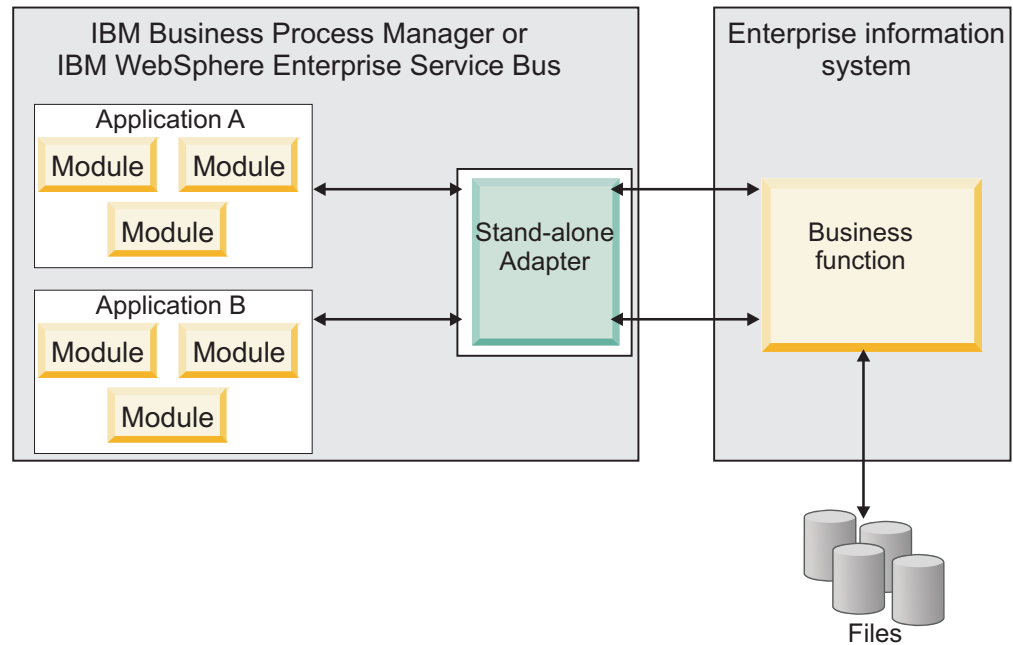
The following are the deployment options:

- **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, as well as 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 given 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 given adapter share the same 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.

Note:

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 runtime, the adapter will return 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 Siebel Business Applications version 7.5, the following error message is displayed: IBM WebSphere Adapter for Siebel Business Applications has loaded 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. When you install multiple adapters with different CWYBS_AdapterFoundation.jar versions, the adapter returns the ResourceAdapterInternalException message, due to a version conflict. To overcome this conflict, you must migrate all adapters to the same version level. For further assistance, contact WebSphere Adapters Support for help.

Considerations while deploying WebSphere Adapter 7.5 with another version

There are occasions when you have to work with embedded adapters that do not need a client-server communication, standalone 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.

Deploying a standalone 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 passthrough inbound operation.
4. Create an application in IBM Integration Designer for a standalone ALE passthrough inbound operation.
5. Install and start the application through the administrative console.
6. Verify the error.

Note: An error message will be 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 will fail. At step 6, you will 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, you may need to generate the artifacts into different folders.

Deploying a combination of standalone and embedded Adapters

1. Install WebSphere Adapter for JDBC version 7.0.1.0 through the IBM 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 standalone 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 standalone and embedded deployments. With this approach, the migration process will handle different CWYBS_AdapterFoundation.jar files and will 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-947.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_Adapters_Family.

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

When you set up a server cluster, you create a Deployment Manager profile. The HAManager, a subcomponent of the Deployment Manager, notifies the Java 2 Platform, Enterprise Edition (J2EE) 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/wasinfo/v8r0/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm_cluster_v61.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. This 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 Siebel application. WebSphere Adapter for Siebel Business Applications 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 2 Platform, Enterprise Edition (J2EE) 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 “Endpoint application of the passive adapter instance listens to the events when `enableHASupport` is set to `True`” on page 121.

Note: In clustered environments, when the adapter works in a Active-Active configuration, it provides both high availability and load balancing support. This functionality is useful in production environments where high performance is needed.

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. Accordingly, if your environment has multiple applications that interact with WebSphere Adapter for Siebel Business Applications 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, as long as 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 those 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 of WebSphere Adapter for Siebel Business Applications

By migrating to version 7.5 of WebSphere Adapter for Siebel Business Applications, 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.

Migration considerations

WebSphere Adapter for Siebel Business Applications version 7.5 may have some features and updates that might affect your existing adapter applications. Before migrating applications that use WebSphere Adapter for Siebel Business Applications, you must consider some factors that might affect your existing applications.

Compatibility with earlier versions

WebSphere Adapter for Siebel Business Applications version 7.5 is fully compatible with the custom business objects (XSD files) and data bindings that are created

using the adapter version 6.1x, version 6.2x, and version 7.0 and enables the existing business objects and data bindings to work well in the latest version of the adapter.

Because version 7.5 of WebSphere Adapter for Siebel Business Applications is fully compatible with version 6.1x, version 6.2x, and version 7.0, any of your applications that used previous versions of WebSphere Adapter for Siebel Business Applications runs unchanged when you upgrade to version 7.5. However, if you want your applications to use features and functionality present in version 7.5 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, or version 7.0 of the adapter with version 7.5 and enables version 7.5 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 of the adapters. If any of your applications embed an adapter that is version 7.0 or earlier and you are upgrading to version 7.5, and you want your applications to take advantage of the features and functions in version 7.5, 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 only supports the migration of adapters from version 6.1x, version 6.2x, and version 7.0 to version 7.5. It does not support the adapter migration from lower versions to any of the versions prior to version 7.5.

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 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 6.1.x, version 6.2.x and version 7.0 to version 7.5, but you do not want to migrate the adapter artifacts, you can do so by deselecting 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 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 of WebSphere Adapter for Siebel Business Applications, 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.

Deprecated features

A deprecated feature is one that is supported but no longer recommended and that might become obsolete. In this version of WebSphere Adapter for Siebel Business Applications, business graphs are now optional. You need a business graph only for modules whose business objects were created in version 6.0.2.

Performing the migration

You can migrate a project or EAR file to version 7.5 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/v7r5mx/topic/com.ibm.wbpm.wid.imuc.doc/topics/tmigscart.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 26.
- 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 29.
- 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.

Note: When you migrate business services in inbound process from version 6.0.2.x, you may have to change the event trigger and ensure that the object key name is same as that existing in the Siebel system. For example, if the object key name in the Siebel Business Applications is "Row Id", the event trigger in Siebel Business Applications should have the same object key, "Row Id" in the event table.

Results

The project or EAR file is migrated to version 7.5. 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 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 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/v7r5mx/topic/com.ibm.wbpm.wid.imuc.doc/topics/tmigrscart.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 Siebel Business Applications, version 7.5.

Migrating WebSphere Business Integration applications

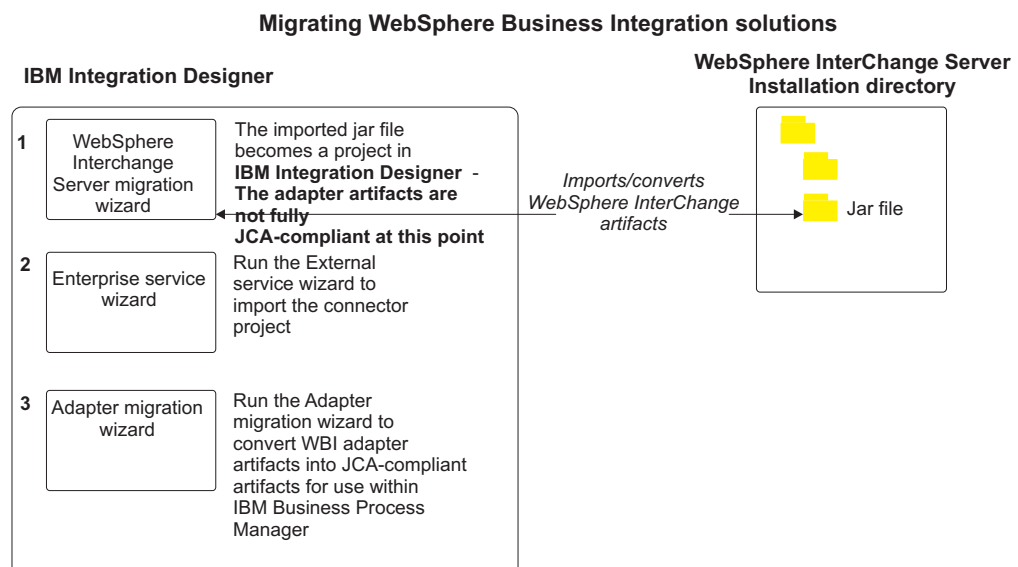
You need to migrate the WebSphere Business Integration applications so that they become compatible with Version 7.5 of your adapter.

About this task

Migrating WebSphere Business Integration applications for use with Version 7.5 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 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 of your adapter.



Migrating applications from WebSphere InterChange Server

To use Version 7.5 of WebSphere Adapter for Siebel Business Applications 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.

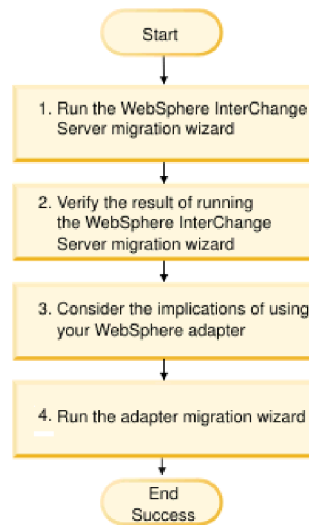


Figure 4. Roadmap for migrating applications from WebSphere InterChange Server

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 of WebSphere Adapter for Siebel Business Applications.
In addition to considerations for migrating WebSphere InterChange Server applications, you need to consider how Version 7.5 of WebSphere Adapter for Siebel Business Applications 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 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 of the adapter.

Migration considerations for WebSphere Business Integration adapters

By migrating to WebSphere Adapter for Siebel Business Applications Version 7.5, you have an adapter that is compliant with the Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) and designed specifically for service-oriented architecture.

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 Siebel Business Applications might require changes to the applications that use Version 7.5 of WebSphere Adapter for Siebel Business Applications, 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/v7r5mx/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 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, you should do the following steps:

- Review the information in “Migration considerations” on page 25.
- 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 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/v7r5mx/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**.

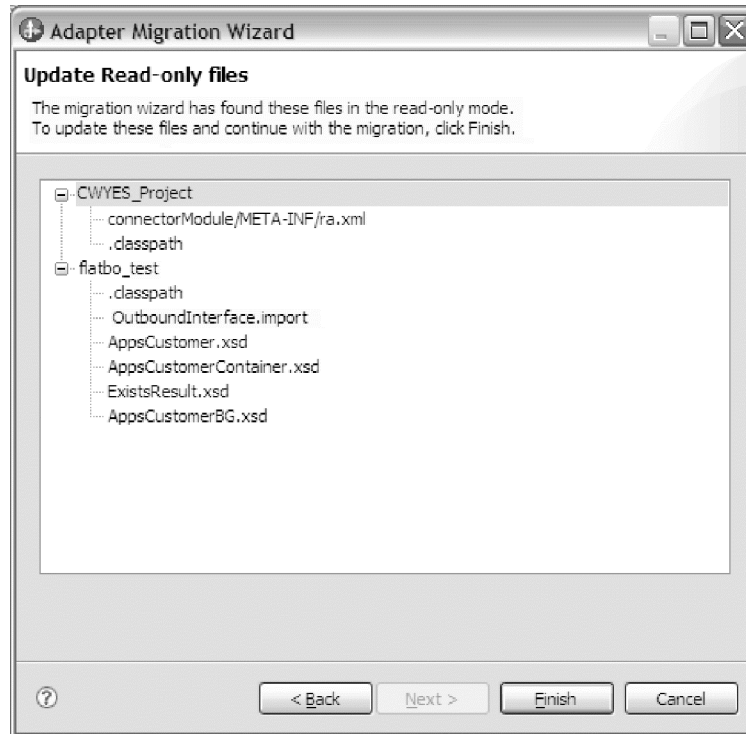
Note:

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. 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 89.

Results

The project is migrated to Version 7.5. 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 adapter specific service description WSDL artifacts. The service description files become JCA compatible. The WSDL files will 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 Siebel eBusiness Applications properties that are either not valid or not supported by WebSphere Adapter for Siebel Business Applications 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://publib.boulder.ibm.com/bpcsamp/index.html>.

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.

Road map for configuring the module

Before you can use WebSphere Adapter for Siebel Business Applications 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 following this road map for the details on how to perform each of these steps.

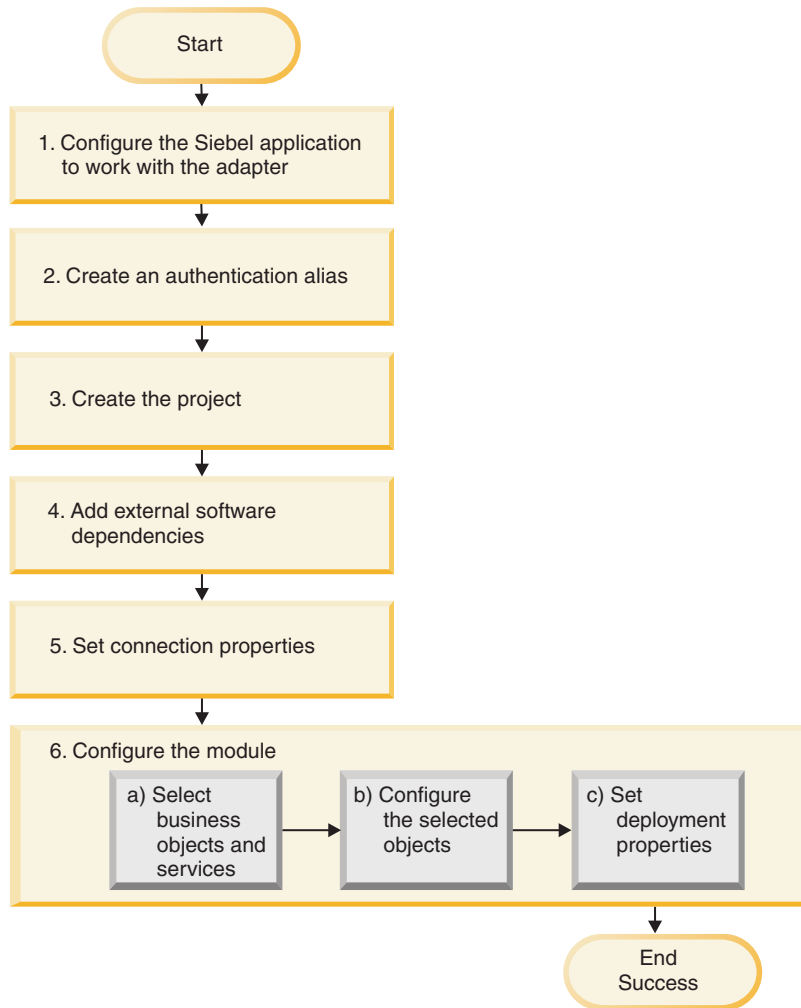


Figure 5. Road map for configuring the module

Configuring the module for deployment environment

This task consists of the following step:

1. Configure the Siebel application to work with your adapter. You perform this step from within the Siebel application.
2. Create an authentication alias to access the Siebel application 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 Siebel Business Applications 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 Siebel application 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 Siebel application, 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 Siebel application 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.

Configuring the Siebel application to work with the adapter

Configure the Siebel application to work with the adapter by creating an event table and a Siebel business object.

Before you begin

Before you configure the Siebel application to work with WebSphere Adapter for Siebel Business Applications, you must create a user name and password so that the external service wizard can connect to Siebel Business Applications to perform outbound operations and retrieve Siebel business objects and services.

Also, because you perform this task on the Siebel Business Applications server, ensure that you are familiar with the Siebel tools required to complete it.

For information about using Siebel tools, refer to the Siebel tools documentation.

You must open Siebel Sales Enterprise (or the Siebel application you are using) on your local database with the administrative privileges.

About this task

To configure the Siebel application, you create an event table and a Siebel business object.

Procedure

1. Create a project called IBM, and lock the project with Siebel tools.
2. Using the object wizard, create an event table called CX_IBM_EVENT to store the events.
 - a. In the event table, create the columns shown in the following table.

Table 4. Event table details

Column Name	Type	Length	Data Type	Required	Nullable	Status
CONNECTOR_ID	Data (public)	255	Varchar	No	Yes	Active
DESCRIPTION	Data (public)	255	Varchar	No	Yes	Active
EVENT_ID	Data (public)	30	Varchar	Yes	No	Active
EVENT_TYPE	Data (public)	20	Varchar	Yes	No	Active

Table 4. Event table details (continued)

Column Name	Type	Length	Data Type	Required	Nullable	Status
OBJECT_KEY	Data (public)	255	Varchar	Yes	No	Active
OBJECT_NAME	Data (public)	255	Varchar	Yes	No	Active
PRIORITY	Data (public)	10	Varchar	No	Yes	Active
STATUS	Data (public)	20	Varchar	Yes	No	Active
XID	Data (public)	255	Varchar	Yes	No	Active

Note: If ConnectorId is specified in the AdapterInstanceEventFilter property, the adapter adds the column to the event query. If ConnectorId is not specified in the AdapterInstanceEventFilter property, the adapter ignores the column in the event query.

- b. Create a business component called IBM Event.
 - c. Create a time stamp called Field Event, and map it to the CREATED column from CX_IBM_EVENT. Make the Type of this field DTYPE_UTCDATETIME.
 - d. Create a business object called IBM Event.
 - e. Associate the IBM event business component to the IBM Event business object.
 - f. Create an applet called IBM Event List Applet, and base it on the IBM Event business component you created earlier.
 - g. Create a view called IBM Event List View, and base it on the IBM Event business object you created earlier.
 - h. Create a screen called IBM Event Screen, and associate it to the IBM Event List View in the Siebel tools.
3. Create a page tab.
 - a. Click **Start Application > Siebel Sales Enterprise > Page tab**.
 - b. Right-click and select **New Record**.
 - c. Specify IBM Event as the screen name and IBM Event for the **Text - String Override** field.
 - d. Leave the Inactive field clear.
 4. Create a business object called Schema Version for your IBM project and associate it to the Schema Version business component.
 - a. Apply the physical schema for the new tables to your local database. You can do this by querying for the new table, 'CX_IBM_EVENT' and selecting the current query to create a physical schema. Leave the table space and index space blank.
 - b. Click **Activate** to activate the new schema.
 5. Add or modify the Siebel VB or e-scripts for the business component that corresponds to the business objects used at your site. Siebel scripts trigger event notification for business objects. Samples are located in the Samples Folder in your adapter installation.
 6. Create a Siebel repository file by compiling the updated and locked projects on your local database. The new repository file will have an extension of .srf.
 7. Open Siebel Sales Enterprise on your local database.
 - a. Create a responsibility called IBM Responsibility for IBM Event List View.
 - b. Add the employees or teams who are responsible for reviewing events to the newly created IBM Responsibility.

- c. Create a user name called IBMCONN (or some other user name to be used by the adapter later). Add the user name to the newly created IBM Responsibility and also to the Administrative Responsibility.
8. Test the application in your local environment to ensure that you have visibility to the IBM Event List View. An event is generated in the view after you create a record in the supported object. As part of the test, create a Account business component instance in Siebel. Confirm that a new Account event appears in the IBM Event List View. (This is assuming that you have added the e-script trigger to the Account business component). If a new Account event does not appear in the view, check for an error and fix it. For more information about the errors that might be generated, check either the Siebel support site or Siebel documentation.
9. When the test performed in Step 8 is successful, add your new and updated projects to your development server.
10. Activate the new table in the development server.
11. Compile a new Siebel repository (.srf) file on the server.
12. Back up the original repository file on the server.
13. Stop the Siebel Business Applications server, and replace the original repository file with the newly created one.
14. Restart the Siebel Business Applications server.

Related concepts

“Event store” on page 8

The event store is a persistent cache where event records are saved until the polling adapter can process them. The adapter uses the event store to track the inbound events.

Creating an authentication alias

The adapter can use an authentication alias to connect to the Siebel application, 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 of IBM Business Process Manager or WebSphere Enterprise Service Bus. You must also know the user name and password to connect to the Siebel application.

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 44.

About this task

An authentication alias is a feature that encrypts the password used by the adapter to access the Siebel application. 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 through IBM Integration Designer, perform the following steps:
 - a. In the Business Integration perspective of Integration Designer, click the **Servers** tab.
 - b. If the server does not show a status of **Started**, right-click the name of the server (for example, **IBM Business Process Manager or WebSphere Enterprise Service Bus**) and click **Start**.
 - c. Right-click the name of the server 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**, click **Java Authentication and Authorization Service > 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 Siebel application 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 have created an authentication alias, which you use when you configure the adapter properties.

Related concepts

“User authentication” on page 18

The adapter supports several methods for supplying the user name and password that are needed to connect to the Siebel application. 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.

Creating the project

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

Before you begin

Ensure that you have gathered the information you need to establish a connection to the Siebel application. For example, you need the name or IP address of the Siebel application and the user ID and password to access it.

About this task

If you have an existing project, you can use it instead of creating a new one. Select it before you start the wizard.

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**. Click **Next**.
2. In the New External Service window, expand the **Adapters** folder, select **Siebel**, and then click **Next**.
3. In the Select an Adapter window, select **IBM WebSphere Adapter for Siebel Business Applications (IBM : *version*)**, where *version* is the version of the adapter you want to use, and then click **Next**.
4. In the Import a RAR File window, accept the default project name in the **Connector project** field or type a different name.
5. In the **Target runtime** field, select the type of server where you want to deploy the module. The wizard creates the artifacts that are appropriate to that server.
6. Click **Next**. The Locate the Required Files and Libraries window is displayed.

Results

A new connector project is created, which contains the adapter RAR file. The project is listed in the Business Integration perspective .

What to do next

Continue working in the external service wizard. The next step is to add database-specific files to the project.

Adding external software dependencies

The external service wizard needs certain files from the Siebel Business Applications server to communicate with it.

Before you begin

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

About this task

To obtain the required files and specify their locations, use the following procedure.

Procedure

1. Obtain the Siebel dependency JAR files from your Siebel Business Applications administrator or the workstation on which the Siebel Business Applications

server is installed. Depending on the version of Siebel Business Applications you use, the software dependencies differ. The following table lists the dependency files needed by supported versions of Siebel Business Applications. The files are shown with the corresponding external service wizard property names.

Table 5. External software dependency files required by Siebel Business Applications

External service wizard property name	Required dependency files for Siebel Business Applications (versions 7.5 and earlier)	Required dependency files for Siebel Business Applications (versions 7.7x, 7.8x, 8.0, and 8.1.1)
Siebel.jar	None	Siebel.jar
SiebelJI_<language code>.jar	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)
SiebelJI_Common.jar	SiebelJI_Common.jar	None

- Copy the files to a local folder on the workstation where you run the external service wizard.
- In the Locate the Required Files and Libraries window in the wizard, specify the location of the files by clicking **Browse** for each property and selecting the location of the file. For example, if you selected the Siebel.jar file, then the location would be: C:\downloads\Siebel8\Siebel.jar.

The following figure shows sample values for the files.

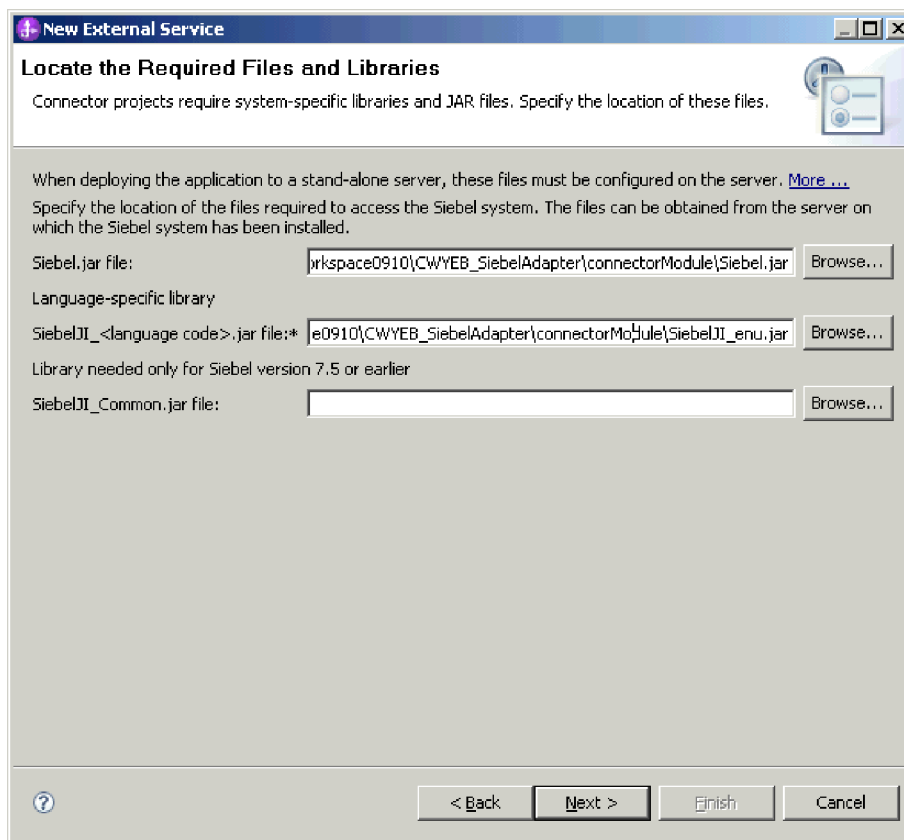


Figure 6. Locate the Required Files and Libraries window, showing the required files selected for Siebel Business Applications, versions 7.7x, 7.8x, and 8.0

- Click **Next**.

Results

The wizard has the files needed to communicate with the Siebel Business Applications server.

What to do next

Continue working in the wizard to configure the module for outbound or inbound processing.

Setting connection properties for the external service wizard

To set connection properties for the external service wizard so that it can access the Siebel Business Applications server, specify information, such as the Connection URL, user name, and the password you use to access the server. In addition, specify the name or IP address of the server.

Before you begin

Make sure that you have successfully added the external software dependency files.

About this task

Specify the connection properties that the external service wizard requires to establish a connection to the Siebel Business Applications 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 **Outbound** (to send data to the Siebel Business Applications server from the adapter), or **Inbound** (to send data to the adapter from the Siebel Business Applications server), and then 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 **Connection URL** field, type the URL to connect to the Siebel Business Applications server.
 - b. Select **Language code** from the drop-down list, to specify the language. The default is set to ENU – English.
 - c. Type the user name and password you use to access the Siebel Business Applications server. The password is case-sensitive.
 - d. In the **Settings for performing discovery on Siebel system** panel, set the **Type of Siebel objects to discover** to Siebel Business Objects or Siebel Business Services from the drop-down list.

Note: If you select Siebel Business Objects, as the **Type of Siebel objects to discover**, the **Prompt for additional configuration settings when adding a business object** check box is enabled.

- e. In the **Siebel repository name** field, type the name of the repository from which the objects are to be discovered. This property is optional.

- f. In the **Prefix for business object names** field, specify a string that can be prefixed to the business object names. As this property is optional, you can leave it blank.
- g. Optional: Select the **Prompt for additional configuration settings when adding a business object** check box to select the attributes of the business objects.

Note: If you have the **Multi Value Link (MVL)** field, it is added by default to the generated business objects.

3. Click **Advanced** to set the additional advanced properties.
 - a. Optional: In the **Advanced properties** area, click **Select** to specify the **Code page** that the adapter uses to connect to the Siebel Business Applications.
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 logging properties for 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 Siebel Business Applications 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 business objects or business services on the Siebel application.

Related reference

“Connection properties for the wizard” on page 133

The external service wizard properties are used to establish a connection between the external service wizard of IBM Integration Designer, a tool that is used to create business objects, and the Siebel Business Applications server. The properties you configure in the external service wizard specify such things as connection configuration, bidirectional (bidi) properties, and logging and tracing options.

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 Siebel application, and to generate the business object definitions and related artifacts.

Related concepts

“Outbound processing” on page 3

WebSphere Adapter for Siebel Business Applications supports synchronous outbound processing. Assume, the component sends a request in the form of a WebSphere business object hierarchy to the adapter. The adapter processes the request and returns a WebSphere business object hierarchy that represents the result of the operation.

Selecting business objects

Select business objects through the external service wizard, that you want to process.

Before you begin

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

About this task

Specify the search criteria that the external service wizard uses to discover business objects on the Siebel Business Applications server.

To search and select business objects, use the following procedure.

Selecting business objects

1. In the Find Objects in the Enterprise System window,
 - a. Click **Edit Query**.
 - b. In the Specify the Query Properties window, type the name of the business object in the **Pattern** field that you want to find (for example, Account), and click **OK**.
 - c. Click **Run Query** to display the specified business object.
 - d. Expand the **Siebel Business Objects** folder in the **Discovered objects** pane to see the list of discovered business objects in the Find Objects in the Enterprise System window.

Note: By default, all the business objects, whether they have a primary business component or not, are displayed.

- e. Expand the required business objects (for example, Account), to see the business components.

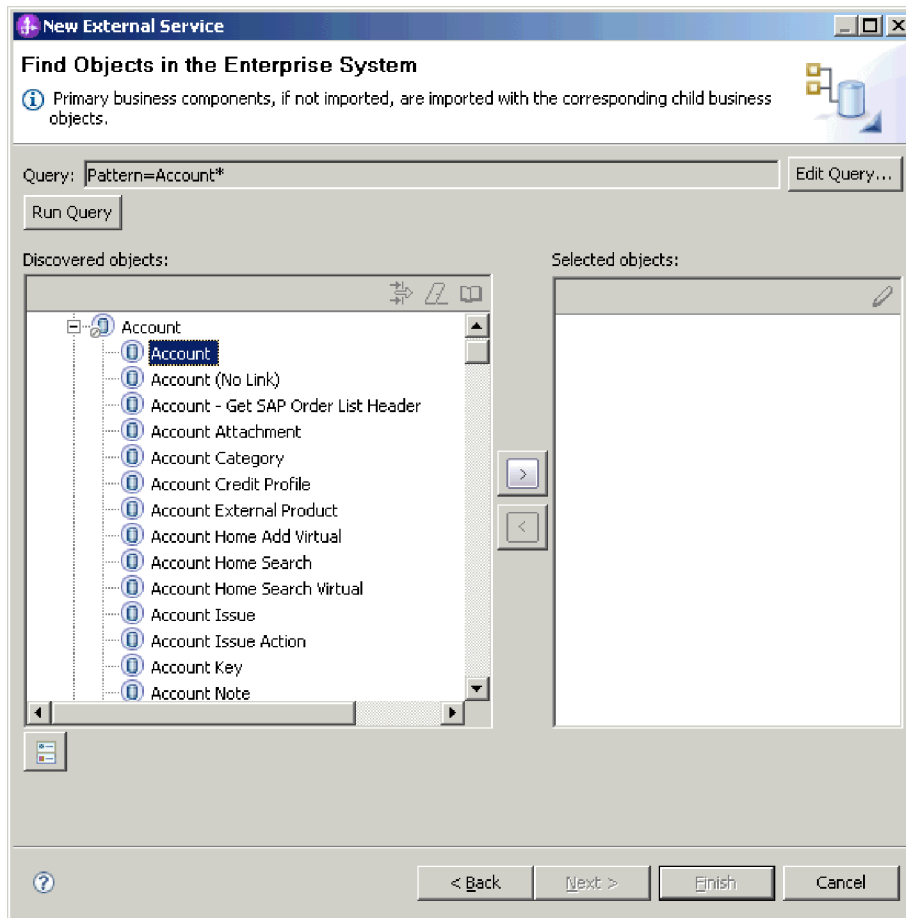


Figure 7. Find Objects in the Enterprise System window, showing the Siebel Business Objects folder expanded with the list of discovered objects

- f. Select the business components you want to work with and click the right arrow to add them to the **Selected objects** pane.

Note: The Specify the Configuration Properties for 'object' window is displayed for the selection of attributes, if you have selected the **Prompt for additional configuration settings when adding a business object** check box in the Specify the Discovery Properties window.

- g. Select the attributes you want to generate from the Specify the Configuration Properties for 'object' window.

Note: If you do not select an attribute, an empty business object is generated.

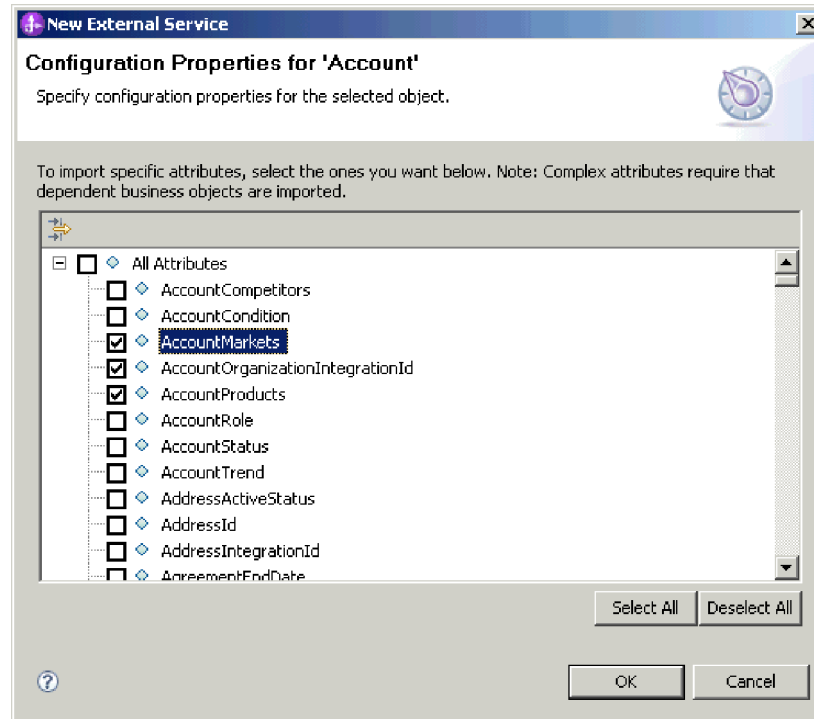


Figure 8. Configuration Properties for 'object' window for selecting and importing the attributes to the Selected Objects pane

- h. In the Find Objects in the Enterprise System window, select the required attributes and click **OK** to add the business components to the Selected objects pane.

Note: You can follow the explained procedure (step a to step g), to continue to select business components from various business objects. For example, you can query on the opportunity business object, which is another business object in Siebel Business Applications server, and select the associated business components.

If a business object has a primary business component in it:

- 1) Select a business component. If this business component is related to the primary business component, the primary business component is selected automatically and operations are generated only for the primary business component. The business object schema is generated for both primary business component and the selected business component.

Note: No operations are generated for the child business components.

- 2) If you select a business component, and if this business component is not related to any primary business component, the operations are generated for this business component along with the business object schema.
- 3) If you select two non-primary business components and if they are related to each other but not related to the primary business component, the operations are generated for each business components along with business object schemas.

When a business object has no primary business component in it:

- 1) Select a business component or a list of business components. The operations are generated for each business component along with the business object schemas.
- i. Click **Next**. The Specify Composite Properties window is displayed.

Results

You have selected the Siebel business object you want to work with and named it. The Specify Composite Properties window is displayed.

What to do next

Configure the objects you have selected.

Related concepts

“Business objects” on page 11

To send data or obtain data from Siebel Business Applications, the adapter 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, an employee record, or an unstructured text.

Related reference

“Business object information” on page 125

A business object is a structure that contains application-specific information (metadata) about how the adapter must process the business object as well as 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.

Selecting business services

Select business services through the external service wizard, to specify the business functions that you want to call .

Before you begin

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

About this task

Specify the search criteria that the external service wizard uses to discover business functions on the Siebel Business Applications server.

To search and select business functions, use the following procedure.

Selecting business services

1. In the Find Objects in the Enterprise System window,
 - a. Click **Edit Query**.
 - b. In the Specify the Query Properties window, type the name of the business object in the **Pattern** field that you want to find (for example, EAI Siebel Adapter), and click **OK**.
 - c. Click Run Query to display the specified business object.
 - d. In the **Discovered objects** pane, expand the **Siebel Business Services** folder to see the names of the business services associated with the specified object.

- e. Expand the name of **Siebel Business Service** you want to work with, and select the business service method (for example, Insert).

Note: You can select more than one method at a time.

- f. Click the right arrow to add the service to the **Selected objects** pane.

Note: If the business service method you selected does not have the integration object already selected, the Specify the Configuration Properties for 'object' window is displayed.

- g. In the Specify the Configuration Properties for 'object' window, select the message associated with the business service method and click **Select** to choose **Integration Object**.

Note: If you select the **Use wrapper for multiple integration objects** check box, then you can choose multiple integration objects for the selected business service to return data. When this option is selected, the adapter generates a wrapper to hold the selected integration objects and converts them during run time. For more information, see “Application-specific information” on page 125.

- h. Click **OK** on Select Value window, and then click **OK** on Specify the Configuration Properties for 'object' window.

Note:

- 1) When a business service has only one complex type input/output argument, the Specify the Configuration Properties for 'object' window is not displayed.
- 2) While you select an input and output complex attribute, even though the integration object cannot be selected, an input and an output SiebelMessage must be specified. Complex attributes are the variables for the SiebelMessage object. There are three types of objects, for example: input, output, and input/output. You must select either an input or an output object or an input/output object.
- 3) A Siebel message I/O type is always displayed in the message node name and you need not click the node to see it.
- 4) An asterisk * is added to the beginning of the Siebel message name if that message is selected.
- 5) If a Siebel message is not selected, then do not select the **Integration Object** and it is not added as a business object.
- 6) Siebel Enterprise Information System (EIS) restricts to create or change the existing input and output argument names for any method of a standard business service. The names are hard-coded internally in the underlying class of the business service along with the logic. Ensure that you are working on a custom Siebel business service if you want to use the custom Siebel message.

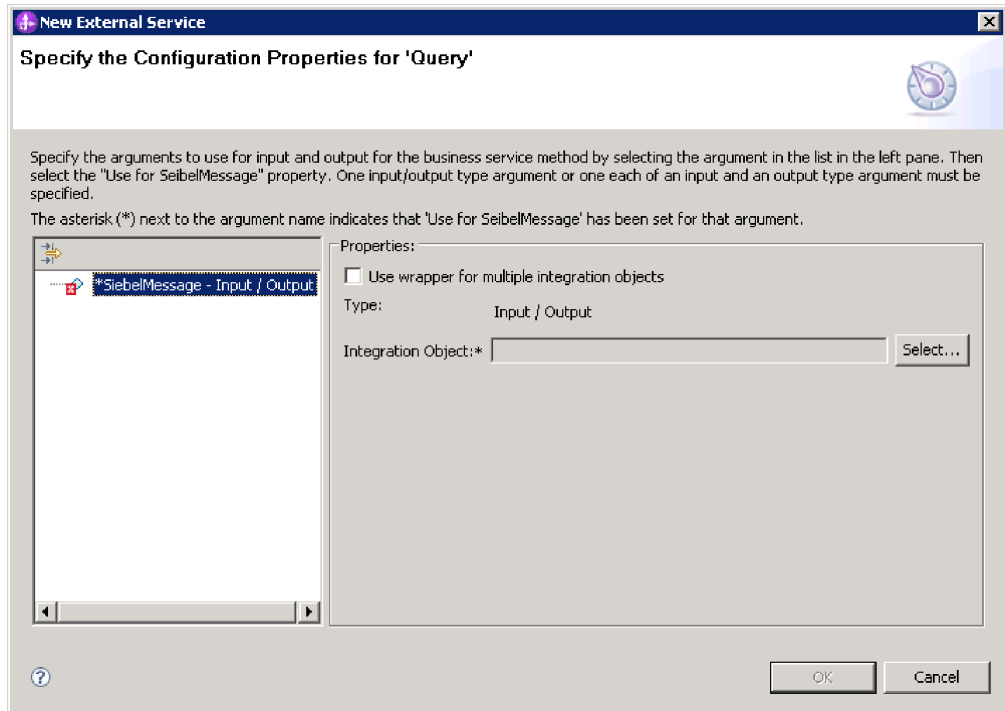


Figure 9. The Configuration Properties for 'object' window specifying the integration value for the business methods

- i. Click Next.

Results

You have selected the Siebel business service you want to work with and named it. The Specify Composite Properties window is displayed.

What to do next

Configure the objects you have selected.

Configuring the selected objects

To configure the business object, you specify information about the object, such as name for the business object and the directory in which it has to be stored.

Before you begin

Make sure that you have selected and imported the business function.

About this task

To configure the business object, use the following procedure.

Procedure

1. In the Specify Composite Properties window of the external service wizard, review the list of operations. This list contains all the operations that the adapter supports for the outbound services for all business objects that you selected in the previous window.

2. In the **Business object namespace** field, accept the default namespace (<http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>) or type the full name of another namespace. If you are adding the business object to an existing module and the module already includes that business object (from an earlier run of the external service wizard), change the namespace value.

For example, you can change the namespace to <http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>.

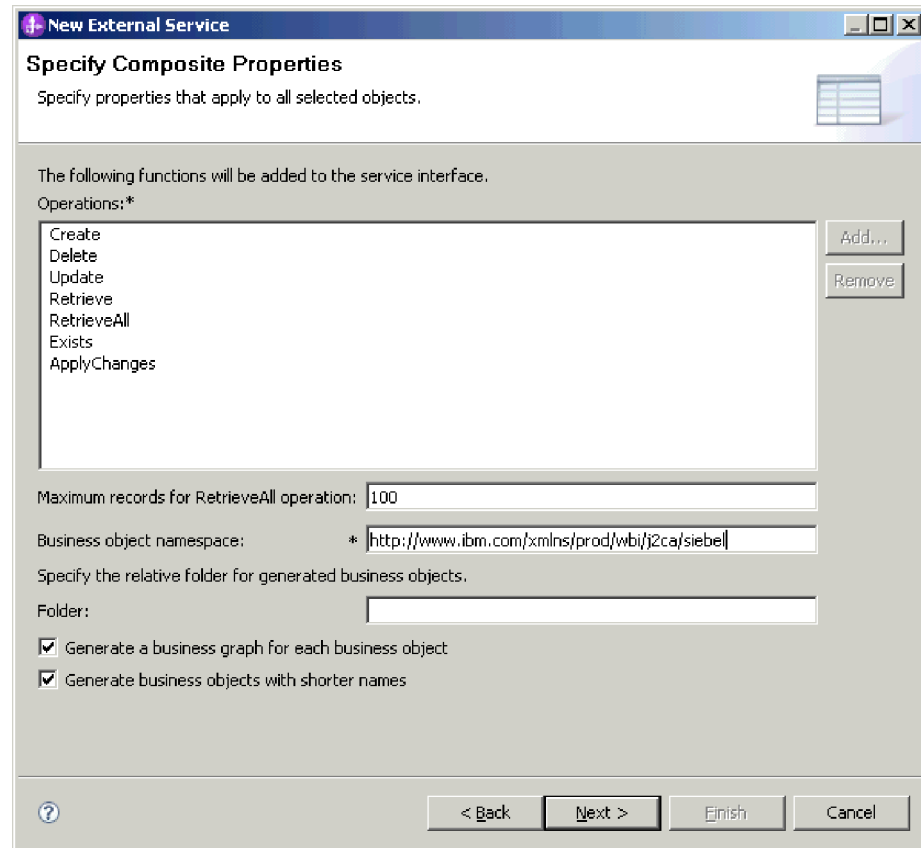


Figure 10. Specify Composite Properties window, showing the default Business object namespace

3. Optionally, in **Folder**, type the relative path to the folder where the generated business objects are to be stored.

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.

4. If you want the business object or business objects to be enclosed within a business graph, select the **Generate a business graph for each business object** check box.
5. If you want the business object or business objects to be generated with shorter names, select the **Generate business objects with shorter names** check box.
6. Click **Next**.

Results

You have specified a name for the top-level business object. If you have completed the optional steps, then you also have specified a location where the business object is stored, whether the business object is enclosed with a business graph, and whether the business object is generated with a shorter name. The Specify the Service Generation and Deployment Properties window is displayed.

What to do next

Continue working in the wizard. The next step is to specify the information for generating the artifacts. Generate a deployable module that includes the adapter and the business objects.

Related concepts

“Business objects” on page 11

To send data or obtain data from Siebel Business Applications, the adapter 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, an employee record, or an unstructured text.

Related reference

“Business object information” on page 125

A business object is a structure that contains application-specific information (metadata) about how the adapter must process the business object as well as 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.

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 Siebel Business Applications. Then, the wizard creates a business integration module where all the artifacts and property values are saved.

Before you begin

Make sure that you have configured the business object. The Specify the Service Generation and Deployment Properties window is displayed.

About this task

Generate the module. The module contains the artifacts that you deployed on the server, which includes the adapter runtime files and the selected business objects. To generate the module, use the following procedure.

Procedure

1. Optionally select **Edit operations** if you want to change the default operation name. In the Edit Operation Names window, type a new name and optional description, and click **OK**.

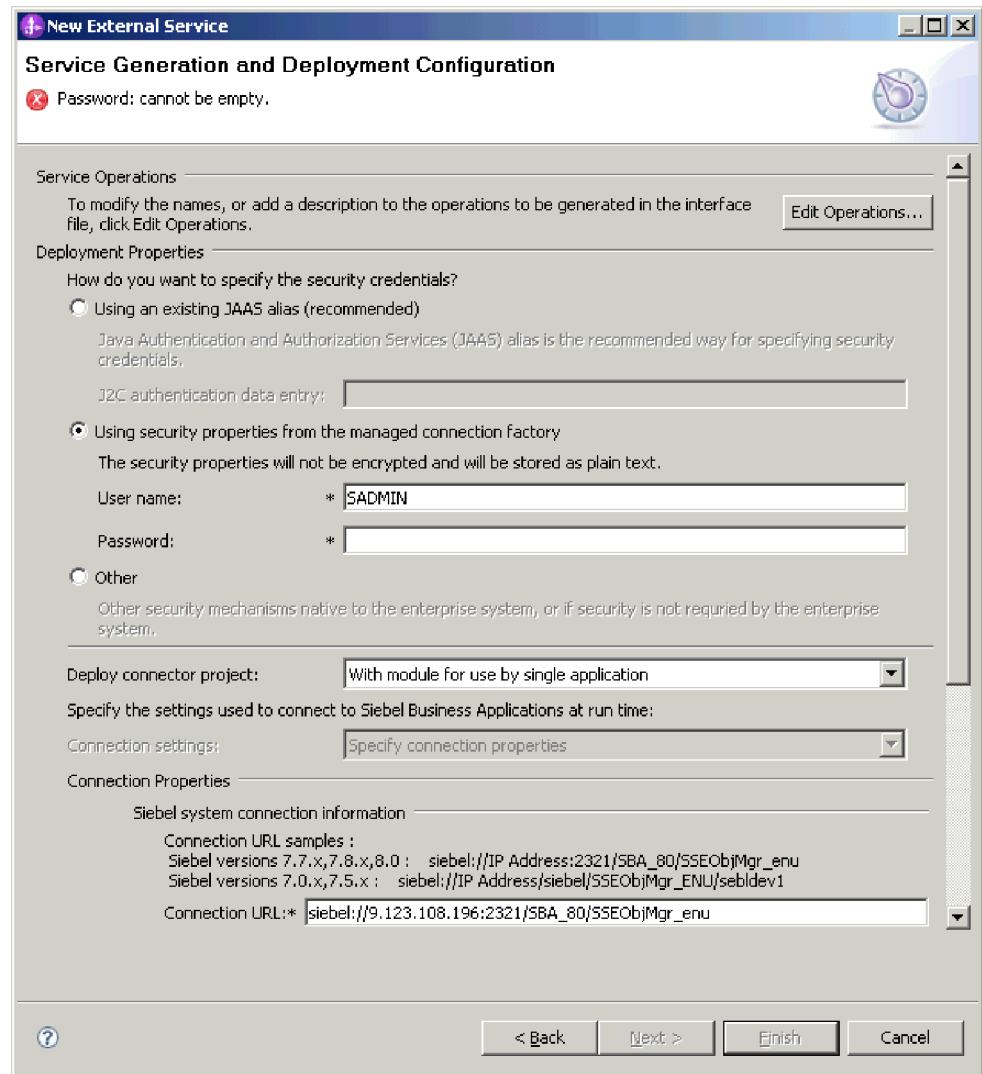


Figure 11. Service Generation and Deployment Configuration window, showing the **Connection properties** section

- 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 the required security credentials in the **Deployment Properties** area:
 - To use a J2C authentication alias, select the **Using an existing JAAS alias (recommended)** field, and specify the name of the alias in the **J2C Authentication Data Entry** field. 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 the **Using security properties from managed connection factory** field, and type the values in the **User name** and **Password** fields.
 - 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:** 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.
4. In the **J2C Authentication Data Entry** field, enter the name you specified in the Security section of the administrative console.
 5. Click **Use discovered connection properties** to set the properties.
If you select **Use connection properties specified on server**, you can configure properties later, using the administrative console.
 6. In the Connection properties section, set or change any connection properties that apply to your configuration. See “Managed connection factory properties” on page 145 for more information about these properties.
Properties marked with an asterisk (*) are required.
 7. Optionally, specify advanced properties by clicking **Advanced**. Expand each of the following advanced sections to review the properties. The following figure shows the advanced property sections on the Specify the Service Generation and Deployment Properties window.

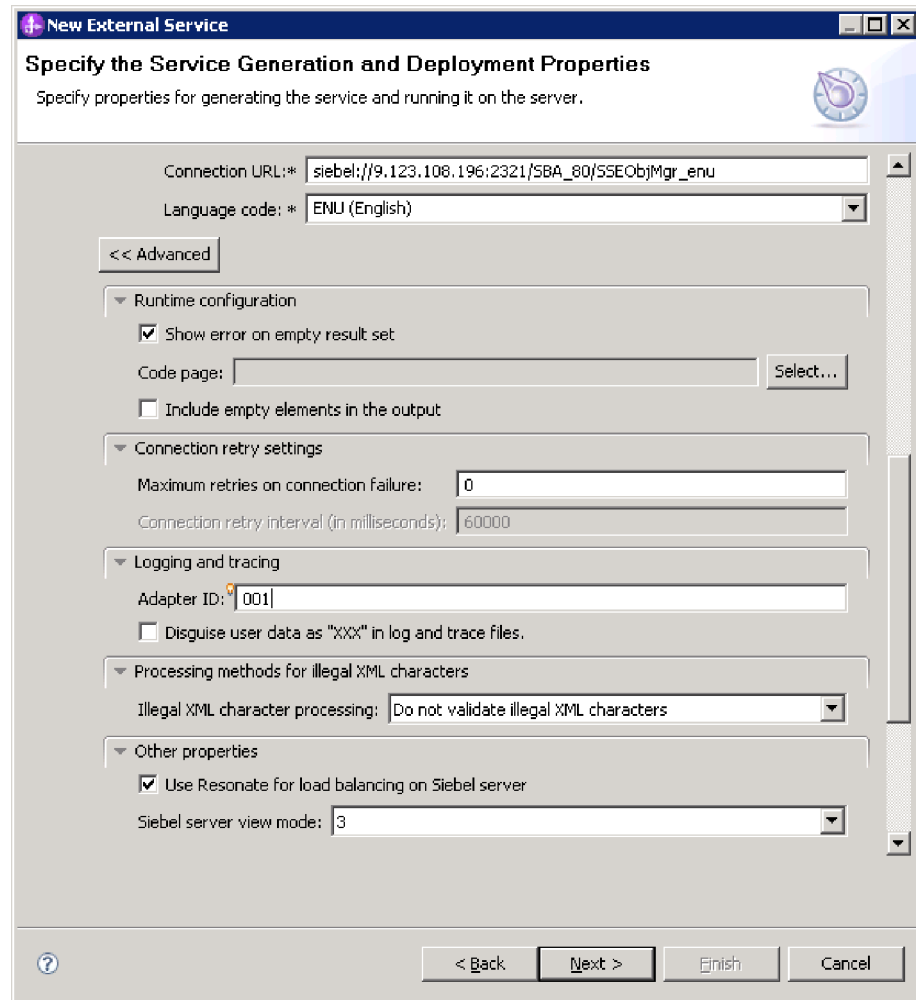


Figure 12. Service Generation and Deployment Configuration window, showing the advanced property sections after clicking the **Advanced** button

See “Managed connection factory properties” on page 145 for more information about these optional properties.

- **Runtime configuration**

- To specify that the adapter throws an exception when no records are found in the RetrieveAll operation for a business object or a business service, select the **Show error on empty result set** check box.
- To specify the code page that the adapter uses to connect to the Siebel Business Applications server, click **Select** and then select the value from the list. For more information, see “Code page (codePage)” on page 147.
- To specify that the adapter returns the elements with an empty string value in the response business object, select the **Include empty elements in the output** check box. For more information, see “Include empty elements in the output (includeEmptyField)” on page 149.

- **Connection retry settings**

- To specify the number of times the adapter can attempt to reconnect to the Siebel Business Applications server in case of connection failure, set **Maximum retries on connection failure** to a value greater than or equal to zero. For more information, see “Maximum retries in case of system connection failure (connectionRetryLimit)” on page 150.

- To specify the time interval between attempts to connect to the Siebel Business Applications server if the connection fails, set **Connection retry interval (in milliseconds)** to a value in milliseconds. This property is enabled only when the connectionRetryLimit property has a value greater than zero. For more information, see “Retry interval if connection fails (in milliseconds) (connectionRetryInterval)” on page 151.
- **Logging and tracing**
 - If you have multiple instances of the adapter, set **Adapter ID** to a value that is unique for this instance.
 - If you want to mask sensitive information in log and trace files (for example, if you want to avoid making customer information visible in these files), select **Disguise user data as "XXX" in log and trace files**.
- **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.

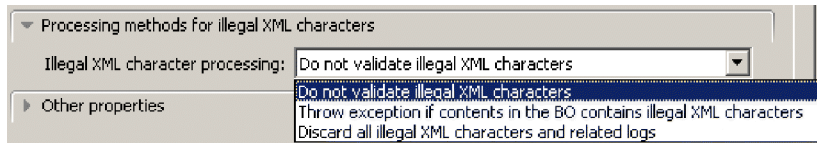


Figure 13. Processing methods for illegal XML characters

- **Other properties**
 - Select **Use Resonate for load balancing on Siebel server** if you want to specify that the Siebel Business Applications server uses resonate support. For more information, see “Use resonate support for load balancing on Siebel server (UseResonateSupportForLoadBalancingOnSiebelServer)” on page 139.
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 you want to create. It can be a new or an existing module.
 - If the required module name is displayed in the **Module** list, select its name.

Note: If the module contains an interface or business object with the same name as any you are now configuring, the original interface or business object in the module can be replaced by the new version.

 - Otherwise, 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, SiebelOutboundModule.
 - d. If you want the service description files (the .import and .wsdl files) to be located 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 automatically opened 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 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. Optionally, 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 the **Name** field, accept the default import name or type a different name.
 13. Optionally, 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. Optionally, in **Description**, type a descriptive comment about the module.
 15. When you finish setting the properties, click **Finish**.

Results

The new module is added to the Business Integration perspective.

What to do next

Export the module as an EAR file for deployment.

Related reference

“Inbound configuration properties” on page 154

WebSphere Adapter for Siebel Business Applications 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.

“Outbound configuration properties” on page 131

IBM WebSphere Adapter for Siebel Business Applications 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.

“Globalization” on page 177

WebSphere Adapter for Siebel Business Applications 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.

“Managed connection factory properties” on page 145

Managed connection factory properties are used by the adapter at run time to create an outbound connection instance with the Siebel application.

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 Siebel application, and to generate business object definitions and related artifacts.

Related concepts

“Inbound processing” on page 8

WebSphere Adapter for Siebel Business Applications supports asynchronous inbound processing. The adapter polls the Siebel applications 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 component.

Selecting business objects

Select business objects through the external service wizard, that you want to process.

Before you begin

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

About this task

Specify the search criteria that the external service wizard uses to discover business objects or business functions on the Siebel application.

To search and select business objects or business functions, use the following procedure.

Selecting business objects

1. In the Find Objects in the Enterprise System window,
 - a. Click **Edit Query**.
 - b. In the Specify the Query Properties window, type the name of the business object in the **Pattern** field that you want to find (for example, Account), and click **OK**.
 - c. Click **Run Query** to display the specified business object.
 - d. Expand the **Siebel Business Objects** folder in the **Discovered objects** pane to see the list of discovered business objects in the Find Objects in the Enterprise System window.

Note: By default, all the business objects, whether they have a primary business component or not, are displayed.

- e. Expand the required business objects (for example, Account), to see the business components.

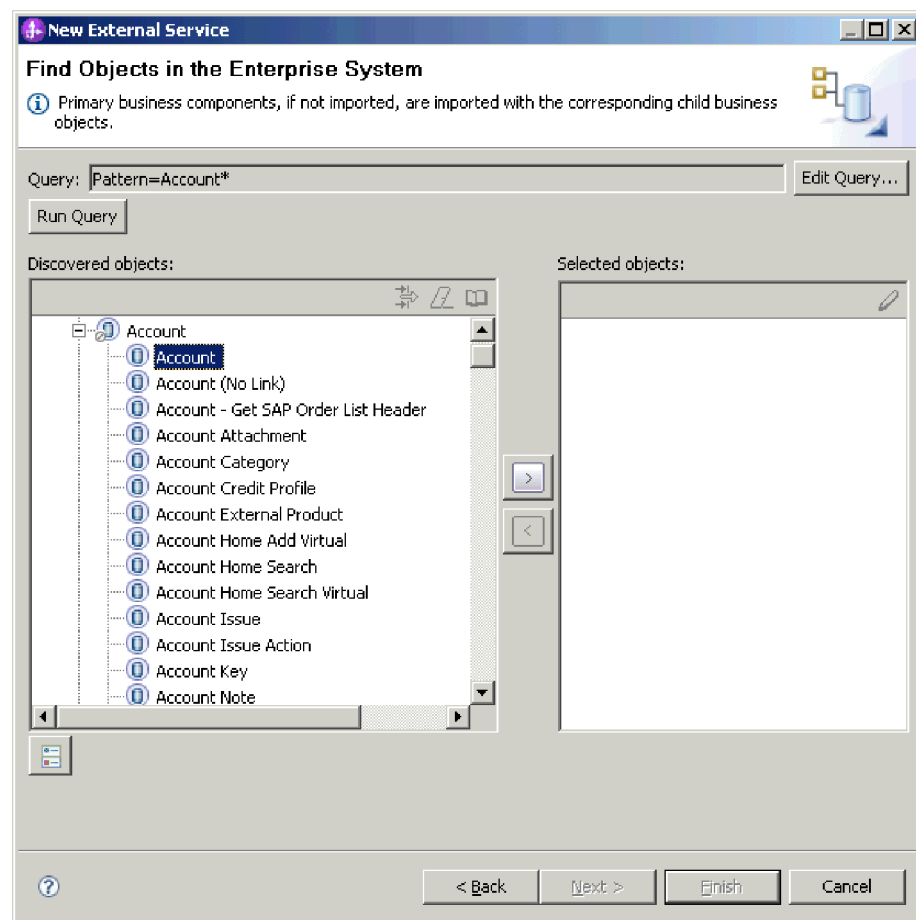


Figure 14. Find Objects in the Enterprise System window, showing the Siebel Business Objects folder expanded with the list of discovered objects.

- f. Select the business components you want to work with and click the right arrow to add them to the **Selected objects** pane.

Note: The Specify the Configuration Properties for 'object' window is displayed for the selection of attributes, if you have selected the **Prompt for additional configuration settings when adding a business object** check box in the Specify the Discovery Properties window.

- g. Select the attributes you want to generate from the Specify the Configuration Properties for 'object' window.

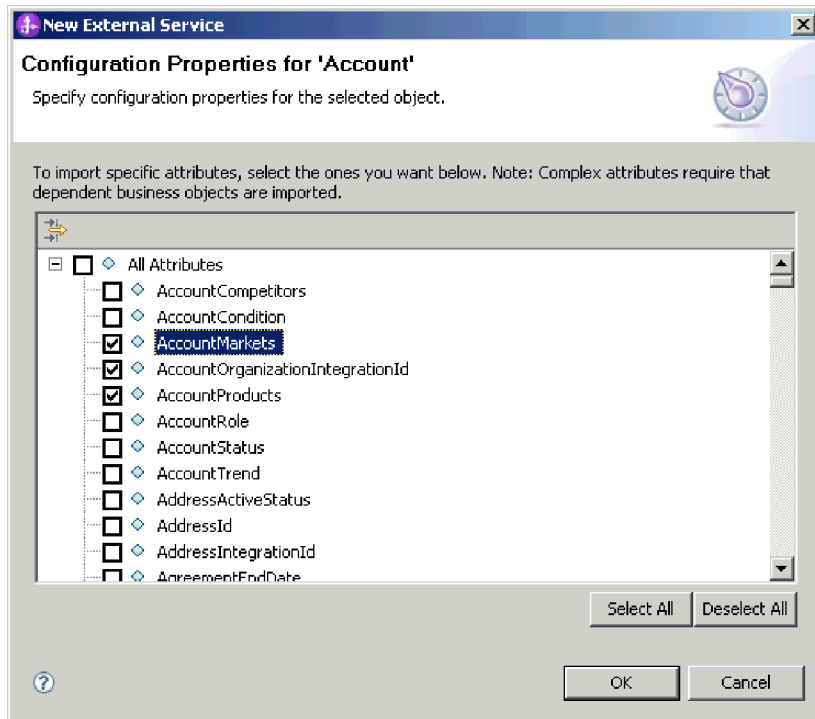


Figure 15. Configuration Properties for 'object' window for selecting and importing the attributes to the Selected Objects pane

Note: If you do not select an attribute, an empty business object is generated.

- h. In the **Discovered objects** window, select the required attributes and click **OK** to add the business components to the **Selected objects** pane.

Note: You can follow the explained procedure (step a to step g), to select the business components from various business objects. For example, you can query on opportunity business objects, which is another business object in Siebel Business Applications server and select its business components.

If a business object has a primary business component in it:

- 1) Select a business component. If this business component is related to the primary business component, the primary business component is selected automatically and operations are generated only for the primary business component. The business object schema is generated for both primary business component and the selected business component.

Note: No operations are generated for the child business components.

- 2) If you select a business component, and if this business component is not related to any primary business component, the operations are generated for this business component along with the business object schema.
- 3) If you select two non-primary business components and if they are related to each other but not related to the primary business component, the operations are generated for each business components along with business object schemas.

When a business object has no primary business component in it:

- 1) Select a business component or a list of business components. The operations are generated for each business component along with business object schemas.
 - i. Click **Next**. The Specify Composite Properties window is displayed.

Results

You have selected the Siebel business object you want to work with and named it. The Specify Composite Properties window is displayed.

What to do next

Configure the objects you have selected. From the Specify Composite Properties window, specify a business object namespace (for example, <http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>) and directory to which the generated business object can be stored. Indicate if you want to generate a business graph for each business object and generate business objects with shorter names.

Related concepts

“Business objects” on page 11

To send data or obtain data from Siebel Business Applications, the adapter 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, an employee record, or an unstructured text.

Related reference

“Business object information” on page 125

A business object is a structure that contains application-specific information (metadata) about how the adapter must process the business object as well as 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.

Selecting business services

Select business services through the external service wizard to specify the business functions that you want to call.

Before you begin

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

About this task

Specify the search criteria that the external service wizard uses to discover business functions on the Siebel application.

To search and select business functions, use the following procedure.

Selecting business services

1. In the Find Objects in the Enterprise System window:
 - a. Click **Edit Query**.
 - b. In the Specify the Query Properties window, type the name of the business object in the **Pattern** field that you want to find (for example, EAI Siebel Adapter), and click **OK**.
 - c. Click Run Query to display the specified business object.
 - d. In the **Discovered objects** pane, expand the **Siebel Business Services** folder to see the names of the business services associated with the specified object.
 - e. Expand the name of the **Siebel Business Service** you want to work with, and select the business service method (for example, Insert).

Note: You can select more than one method at a time.

- f. Click the right arrow to add the service to the **Selected objects** pane.

Note: If the business service method you selected does not have the integration object already selected, the Specify the Configuration Properties for 'object' window is displayed.

- g. In the Specify the Configuration Properties for 'object' window, select the message associated with the business service method and click **Select** to choose the **Integration Object**.
- h. Click **OK** on Select Value window, and then click **OK** on Specify the Configuration Properties for 'object' window.

Note:

- 1) When a business service has only one complex type Input/Output argument, the Specify the Configuration Properties for 'object' window is not displayed.
- 2) While you select an input and output complex attribute, even though the integration object cannot be selected, an input and output SiebelMessage must be specified. Complex attributes are the variables for the SiebelMessage object. There are three types of objects, for example: input, output, and input/output. You must select either an input or an output object or an input/output object.
- 3) A Siebel message I/O type is always displayed in the message node name and you need not click the node to see it.
- 4) An asterisk * is added to the beginning of the Siebel message name if that message is selected.
- 5) If a Siebel message is not selected, then do not select **Integration Object** and it is not added as a business object.
- 6) Siebel Enterprise Information System (EIS) restricts to create or change the existing input and output argument names for any method of a standard business service. The names are hard-coded internally in the underlying class of the business service along with the logic. Ensure that you are working on a custom Siebel business service if you want to use the custom Siebel message.

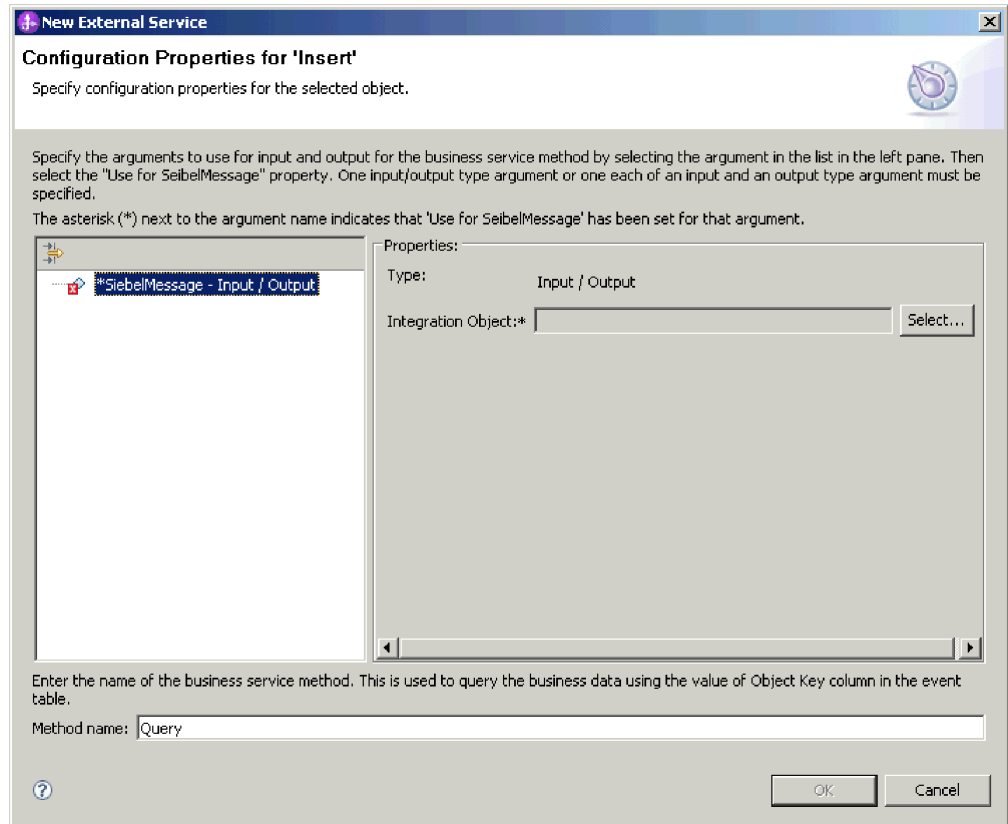


Figure 16. Configuration Properties window specifying the integration value for the business methods

- i. Click Next.

Results

You have selected the Siebel business service you want to work with and named it. The Specify Composite Properties window is displayed.

What to do next

Configure the objects you have selected. From the Specify Composite Properties window, specify a business object namespace (for example, <http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel>) and directory to which the generated business object can be stored. Indicate if you want to generate a business graph for each business object and generate business objects with shorter names.

Configuring the selected objects

To configure the business object, you specify information about the object, such as a name for the business object and the directory in which it must be stored.

Before you begin

Ensure that you have selected and imported the business function.

About this task

To configure the business object, use the following procedure.

Procedure

1. In the Specify Composite Properties window of the external service wizard, review the list of operations. This list contains all the operations that the adapter supports for the inbound services for all business objects that you selected in the previous window. The specified operations are set for all business objects that are generated.
2. In the **Business object namespace** field, accept the default namespace (`http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel`) or type the full name of another namespace. If you are adding the business object to an existing module and the module already includes that business object (from an earlier run of the external service wizard), change the namespace value.

For example, you could change the namespace to `http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel`

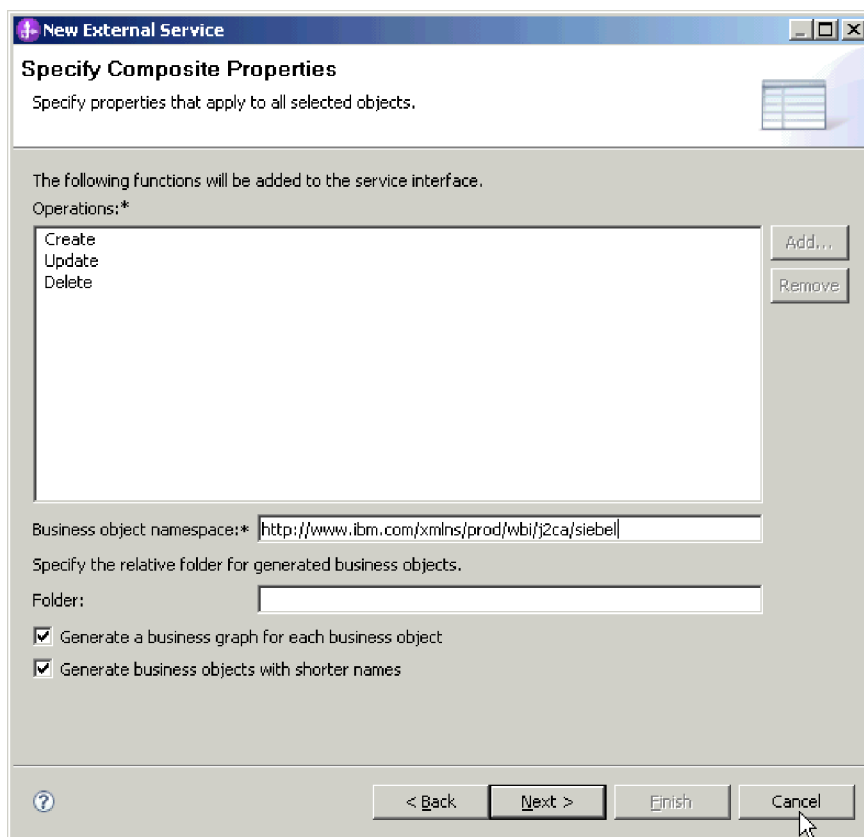


Figure 17. Specify Composite Properties window, showing the default Business object namespace

3. Optionally, in **Folder**, type the relative path to the folder where the generated business objects are to be stored.

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.

4. If you want the business object or business objects to be enclosed within a business graph, select **Generate a business graph for each business object** check box.
5. If you want the business object or business objects to be generated with shorter names, select **Generate business objects with shorter names**, and then Click **Next**.

Results

You specified a name for the top-level business object. If you completed the optional steps, then you also specified a location where the business object is stored, whether the business object is enclosed with a business graph, and whether the business object is generated with a shorter name. The Specify the Service Generation and Deployment Properties window is displayed.

What to do next

Continue working in the wizard. The next step is to specify the information for generating the artifacts. Generate a deployable module that includes the adapter and the business objects.

Related concepts

“Business objects” on page 11

To send data or obtain data from Siebel Business Applications, the adapter 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, an employee record, or an unstructured text.

Related reference

“Business object information” on page 125

A business object is a structure that contains application-specific information (metadata) about how the adapter must process the business object as well as 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.

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 Siebel Business Applications. Then, the wizard creates a business integration module where all the artifacts and property values are saved.

Before you begin

Make sure that you have configured the business object. The Specify the Service Generation and Deployment Properties window is displayed.

About this task

Generate the module which includes the adapter and configured business object. The module is the artifact you deploy on the server.

To generate the module, use the following procedure.

Procedure

1. Optional: Select **Edit operations** if you want to change the default operation name. Then, in the Edit Operation Names window, type a new name and optional description, and click **OK**.

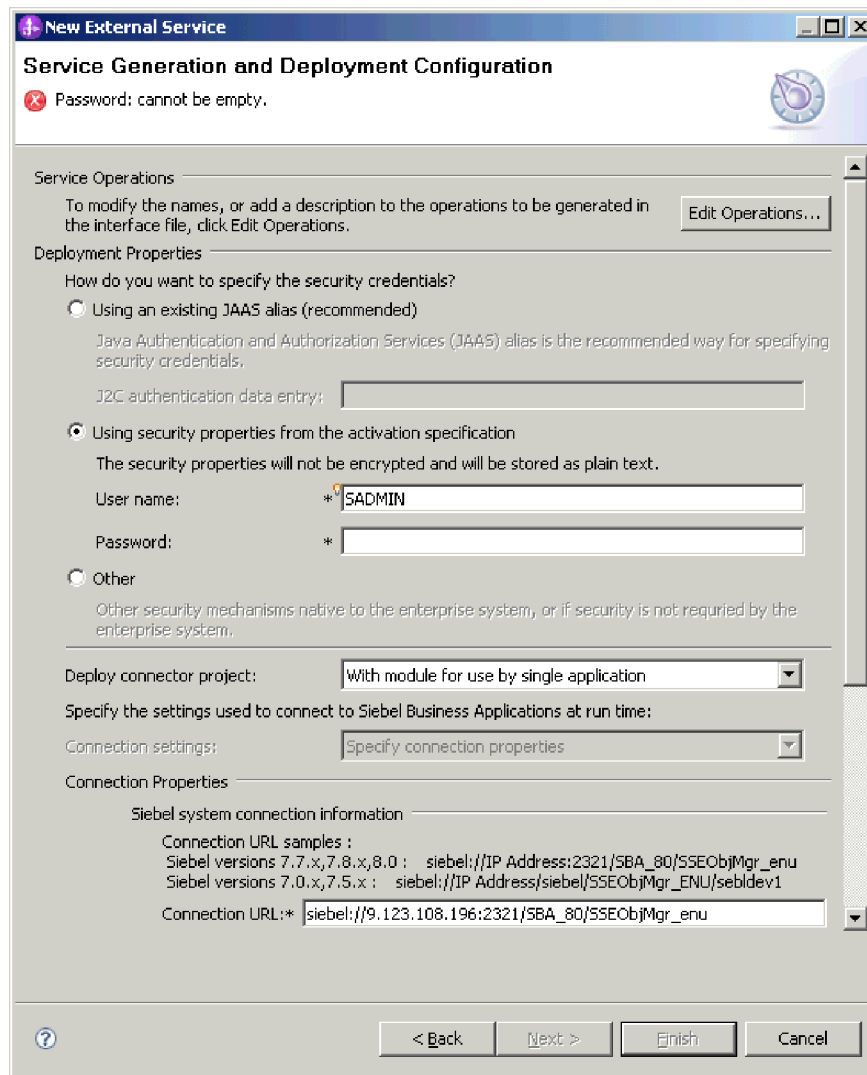


Figure 18. Service Generation and Deployment Configuration window, showing the **Connection properties** section

- 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 the required security credentials in the **Deployment Properties** area:
 - To use a J2C authentication alias, select the **Using an existing JAAS alias (recommended)** field, and specify the name of the alias in the **J2C Authentication Data Entry** field. 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 the **Using security properties from managed connection factory** field, and type the values in the **User name** and **Password** fields.
 - 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:** 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.
 4. In the **J2C Authentication Data Entry** field, enter the name you specified in the Security section of the administrative console.
 5. Click **Use discovered connection properties** to set properties at this time. If you select **Use connection properties specified on server**, you can configure properties later, using the administrative console.
 6. In the Connection properties section, set or change any connection properties that apply to your configuration. See “Activation specification properties” on page 168 and “Connection properties for the wizard” on page 133 for more information about these properties.
Properties marked with an asterisk (*) are required.
 7. Optional: Specify advanced properties by clicking **Advanced**. Expand each of the following group sections to review the advanced properties:
 - Event polling configuration
 - Event delivery configuration
 - Event configuration
 - Runtime configuration
 - Logging and tracing properties
 - Other properties

The following figure shows the advanced property sections on the Specify the Service Generation and Deployment Properties window.

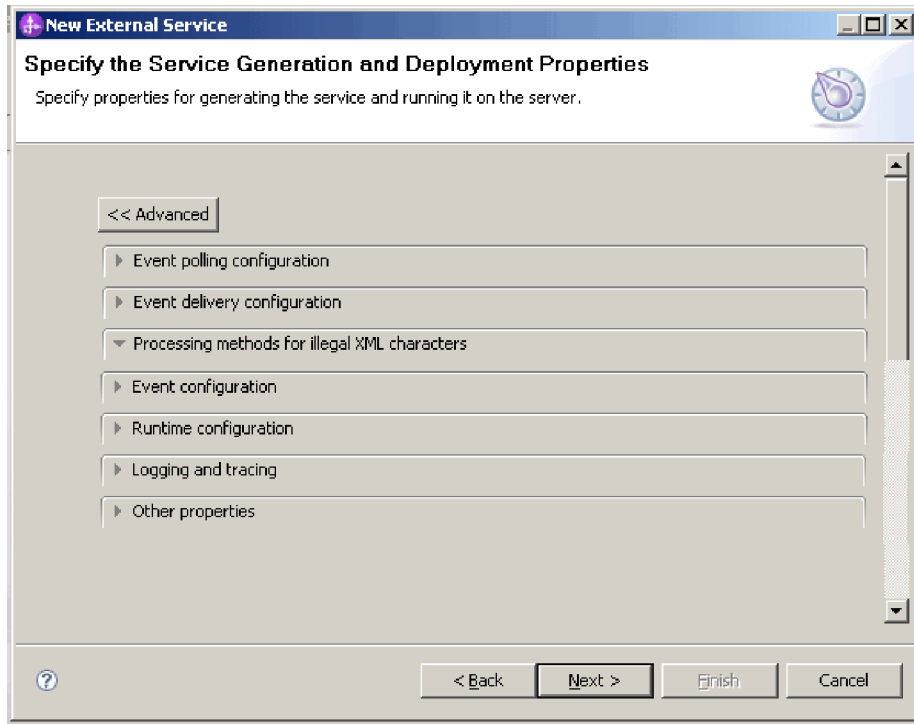


Figure 19. Service Generation and Deployment Configuration window, showing the advanced property sections after clicking **Advanced**

Note: See “Activation specification properties” on page 168 and “Resource adapter properties” on page 163 for more information about these optional properties.

- **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 173.
- 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 174.
- 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 “Retry interval if connection fails (retryInterval)” on page 176.
- d. In **Number of times to retry the system connection**, type the number of times to retry the connection before reporting a polling error. For more information, see “Number of times to retry the system connection (retryLimit)” on page 175.
- 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 177.
- f. You can select **Retry EIS connection on startup** if you want the adapter to retry an inbound connection that was not made to the Siebel application when starting. Only communication failures to the Siebel application are considered. For more information, see “Retry EIS connection on startup (retryConnectionOnStartup)” on page 176.

g. Polling based on calendar:

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.

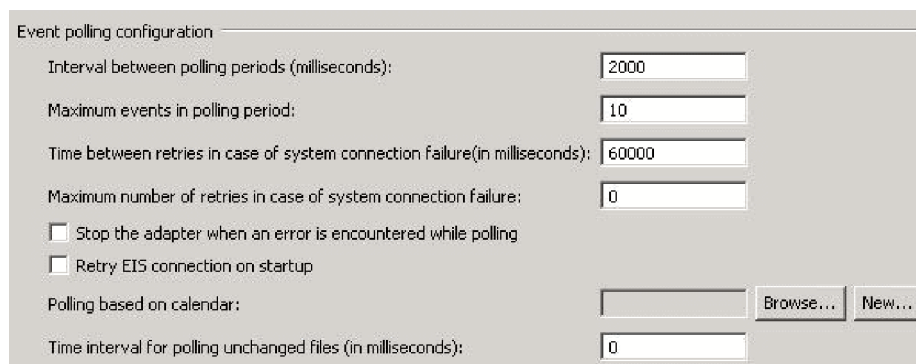


Figure 20. Polling based on calendar

You can either select a blank calendar or create a new calendar for a module or library. When you select a blank calendar, you will not be able to set pre-defined time intervals. You have to define your time intervals. When you create a calendar using a pre-defined template, you can define time intervals for each template.

- 1) Click **New** to create a new calendar entry for a module or library.
You can choose an existing calendar, or create a new calendar instance.
 - Click **Browse** to select an existing calendar module. Or click **New** to create a module for the new calendar.
 - Click **Browse** to choose a folder for the calendar. (Optional).
 - Enter a name for the new calendar.
 - Click **Next** if you want to generate the calendar, through a predefined template. Or, click **Finish**, to create a non template calendar.
- 2) Click **Browse** to select an existing calendar for a module or library. In the **Select a Business Calendar** screen you can search for all currently existing calendar files (*.cal) in the IBM Integration Designer workspace.
 - In the **Name** field, type the calendar name or click the calendar in the **Matching business calendars** screen. Click **OK** to open the external service wizard.
 - In the **WebSphere Integration workspace**, select the **Calendar** module, and browse **Integration logic->Calendars**, to view or modify the calendar schedules. You can modify the intervals and exceptions, or add new entries for these elements. For more information, refer to the information at <http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=/com.ibm.wbpm.wid.bpel.doc/topics/cbuscal.html>.

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**

- In **Type of delivery**, select the delivery method. The methods are described in “Delivery type (deliveryType)” on page 170.
- 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 171.

- 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 172.

- 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 175 and “Maximum connections (MaximumConnections)” on page 174.

- **Processing methods for illegal XML characters**

- 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.

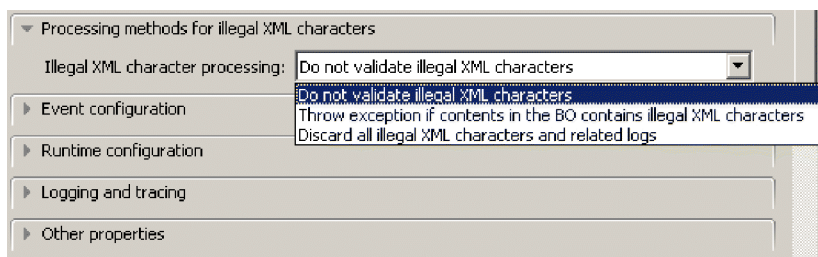


Figure 21. Processing methods for illegal XML characters

- **Event configuration**

- In **Siebel business object name for event store** field, specify the name of the business object in the event store where events are stored for inbound processing. For more information, see “Siebel business object name for event store (SiebelBusinessObjectNameForEventStore) ” on page 138.

Note: You cannot use multiple adapters for polling the same event.

- **Runtime configuration**

- In **Code page** field, specify the code page that the adapter uses to connect to the Siebel Business Applications server by clicking **Select**. Select the value from the displayed list. For more information, see “Code page (codePage)” on page 170.
- Select the **Include empty elements in the output** check box for the adapter to include the elements with empty string value in the response business object. For more information, see “Include empty elements in the output (includeEmptyField)” on page 173

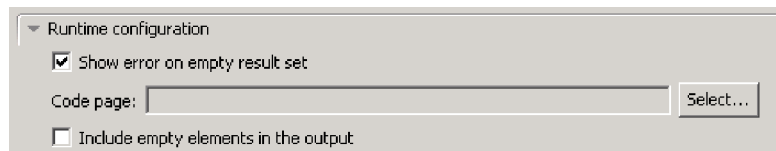


Figure 22. Runtime configuration

- **Logging and tracing**

- If you have multiple instances of the adapter, set **Adapter ID** to a value that is unique for this instance.
- If you want to mask sensitive information in log and trace files (for example, if you want to avoid making customer information visible in these files) expand **Logging and tracing**, provide an ID for the adapter instance (or accept the default value), and select **Disguise user data as "XXX" in log and trace files**.

- **Other properties**

- In **Delimiter for event store**, specify the delimiter used between two name value pairs containing the object key name and value. For more information, see “Delimiter for keys in the event store (DelimiterForKeysInTheEventStore) ” on page 136.
- Select **Use Resonate for load balancing on Siebel server** if you want to specify that the Siebel server uses resonate support. For more information, see “Use resonate support for load balancing on Siebel server (UseResonateSupportForLoadBalancingOnSiebelServer) ” on page 139.

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 you want to create. It can be a new or an existing module.

- If the required module name appears in the **Module** list, select its name.

Note: If the module contains an interface or business object with the same name as any you are now configuring, the original interface or business object in the module will be replaced by the new version.

- Otherwise, 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, SiebelInboundModule.
 - d. If you want the service description files (the `.import` and `.wsdl` files) to be located 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 automatically opened 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 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 the **Name** field, 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 the **Description** field, type a descriptive comment about the module.
 15. When you finish setting the properties, click **Finish**.

Results

The new module is added to the Java EE perspective.

What to do next

Export the module as an EAR file for deployment.

Related reference

“Inbound configuration properties” on page 154

WebSphere Adapter for Siebel Business Applications 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.

“Outbound configuration properties” on page 131

IBM WebSphere Adapter for Siebel Business Applications 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.

“Globalization” on page 177

WebSphere Adapter for Siebel Business Applications 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.

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 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 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” on page 49.
 - a. To search for and select business objects or business functions, see “Selecting business objects” on page 49.
 - b. To modify the configuration of an object from the **Selected objects** list, select the object name and then click the  (Edit) icon.
 - c. 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 54.
5. Click **Next**.
6. In the Service Generation window, modify the service operations if required.
7. Click **Finish**. The artifacts are regenerated.

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 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 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” on page 62.
 - a. To search for and select business objects or business functions, see “Selecting business objects” on page 62.
 - b. To modify the configuration of an object from the **Selected objects** list, select the object name and then click the  (Edit) icon.
 - c. 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 67.
5. Click **Next**.
6. In the Service Generation window, modify the service operations if required.
7. Click **Finish**. The artifacts are regenerated.

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.

Related reference

“Interaction specification properties” on page 153

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 assembly editor in IBM Integration Designer. The properties reside in the method binding of the import.

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 dependencies

Dependent JARs have to be added to the libraries directory or packaged in the EAR.

About this task

The JARs are set in the class path and these dependent libraries have to be made available for run time when the module is deployed. There are two ways to make the dependent libraries available, one for either stand-alone deployment or embedded deployment and the other for embedded deployment only.

Adding external software dependencies on the server

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

About this task

You do not need to perform this task if the Siebel application is installed on the same computer system as IBM Business Process Manager or WebSphere Enterprise Service Bus. The files are already available to the adapter.

Procedure

1. Obtain the Siebel dependency JAR files from your Siebel Business Applications administrator or the workstation on which the Siebel server is installed. Depending on the version of Siebel Business Applications you use, the software dependencies differ. The following table lists the dependency files needed by supported versions of Siebel Business Applications. Each file is shown with the corresponding field label in the external service wizard.

Table 6. External software dependency files required by Siebel Business Applications

External service wizard property name	Required dependency files for Siebel Business Applications (versions 7.5 and earlier)	Required dependency files for Siebel Business Applications (versions 7.7x, 7.8x, 8.0, and 8.1.1)
Siebel.jar	None	Siebel.jar
SiebelJI_<language code>.jar	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)
SiebelJI_Common.jar	SiebelJI_Common.jar	None

2. Copy the files to the server.
 - In a testing environment in 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.

Adding external software dependencies when the adapter is bundled

You must copy the dependent JAR files to the EAR application before you can run your adapter applications. You must use this method only for embedded deployment.

About this task

To obtain the required files and copy them to the EAR application, use the following procedure:

Procedure

1. From the appropriate module, go to the workspace and copy the JAR files to the directory. For example if the name of the module is `ModuleName`, then go to the workspace and copy the JAR files to the `ModuleNameApp/EarContent` directory.
2. Modify the adapter RAR's manifest file, `manifest.mf`, with the list of JAR files required by the adapter. Add the JAR files in the following format: `Class-Path: dependantjar1.jar, dependantjar2.jar`.
3. Copy the native libraries to the run time bin directory and deploy the application.

Results

The vendor software libraries are now a part of your run time environment.

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 the Siebel JAR files installed on the IBM Business Process Manager or WebSphere Enterprise Service Bus server to be able to communicate with the Siebel application.

About this task

You do not need to perform this task if the Siebel application is installed on the same computer system as IBM Business Process Manager or WebSphere Enterprise Service Bus. The files are already available to the adapter.

Procedure

1. Obtain the Siebel dependency JAR files from your Siebel Business Applications administrator or the workstation on which the Siebel server is installed. Depending on the version of Siebel Business Applications you use, the software dependencies differ. The following table lists the dependency files needed by supported versions of Siebel Business Applications. Each file is shown with the corresponding field label in the external service wizard.

Table 7. External software dependency files required by Siebel Business Applications

External service wizard property name	Required dependency files for Siebel Business Applications (versions 7.5 and earlier)	Required dependency files for Siebel Business Applications (versions 7.7x, 7.8x, 8.0, and 8.1.1)
Siebel.jar	None	Siebel.jar
SiebelJI_<language code>.jar	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)	SiebelJI_<language code>.jar (for example, SiebelJI_enu.jar)
SiebelJI_Common.jar	SiebelJI_Common.jar	None

2. Copy the files to the server.
 - In a testing environment in 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 2 Connector (J2C) architecture.

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 J2EE 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**.

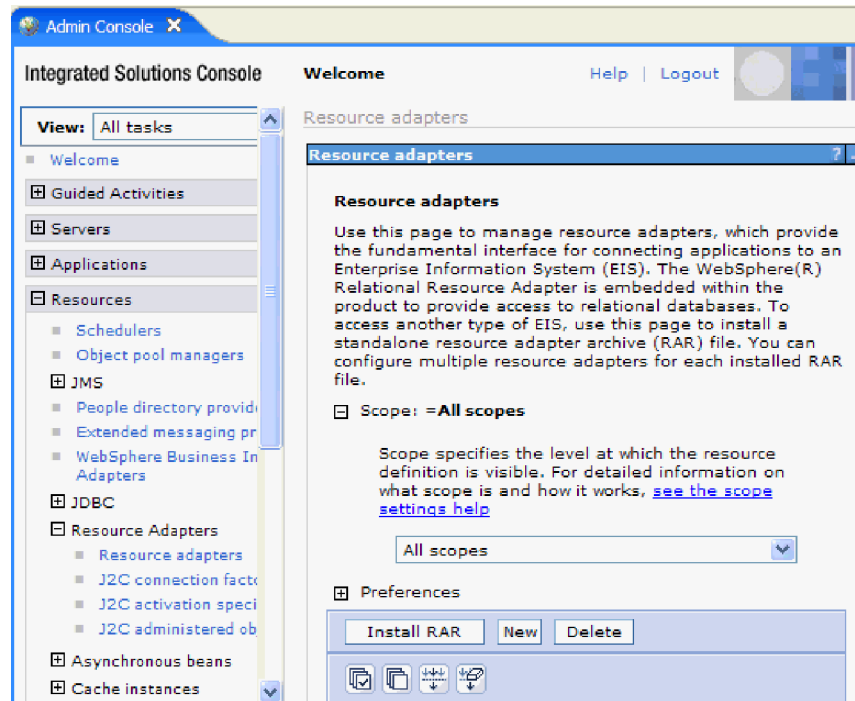


Figure 23. The Install RAR button on the Resource adapters page

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. 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. 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 24. 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.

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.

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).

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.

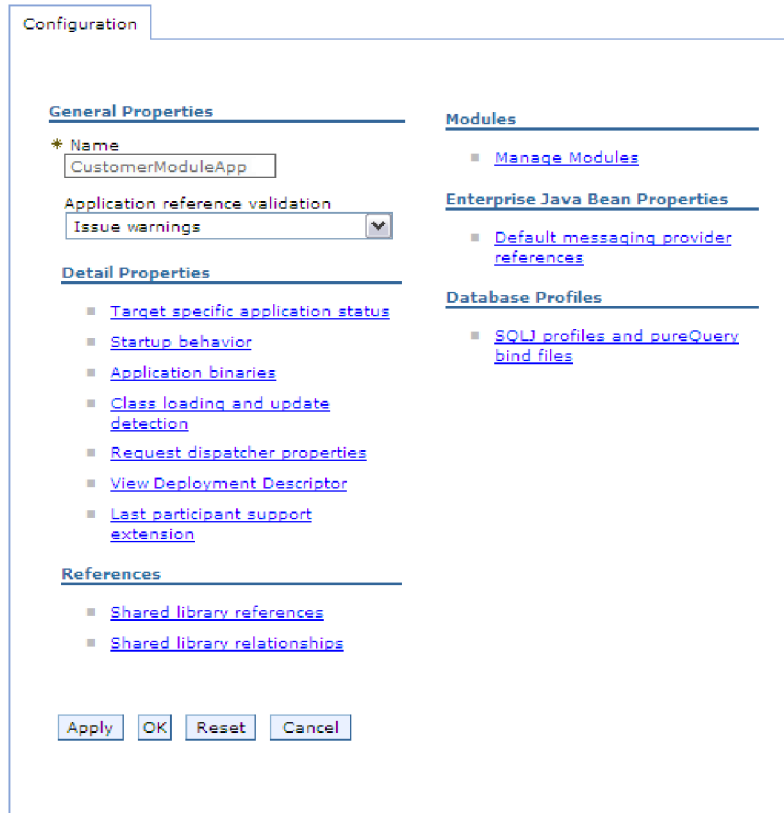


Figure 25. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for Siebel Business Applications**.
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 140 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.

Related reference

“Resource adapter properties” on page 140

The resource adapter properties control the general operation of the adapter. You set the resource adapter properties in the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

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 Siebel application 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.

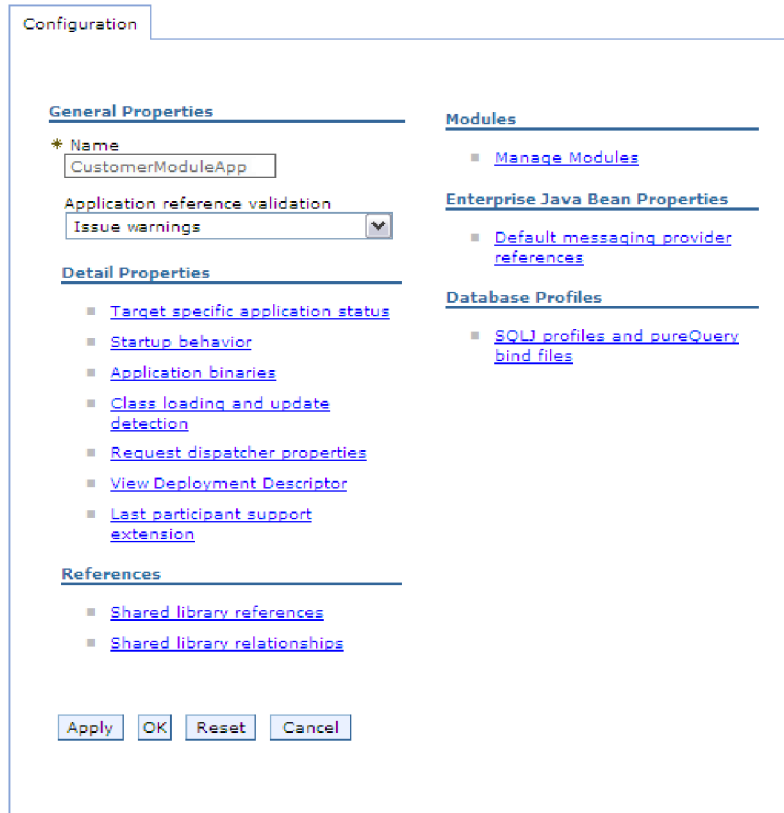


Figure 26. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for Siebel Business Applications**.
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 Siebel Business Applications. 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 145 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.

Related reference

“Managed connection factory properties” on page 145

Managed connection factory properties are used by the adapter at run time to create an outbound connection instance with the Siebel application.

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.

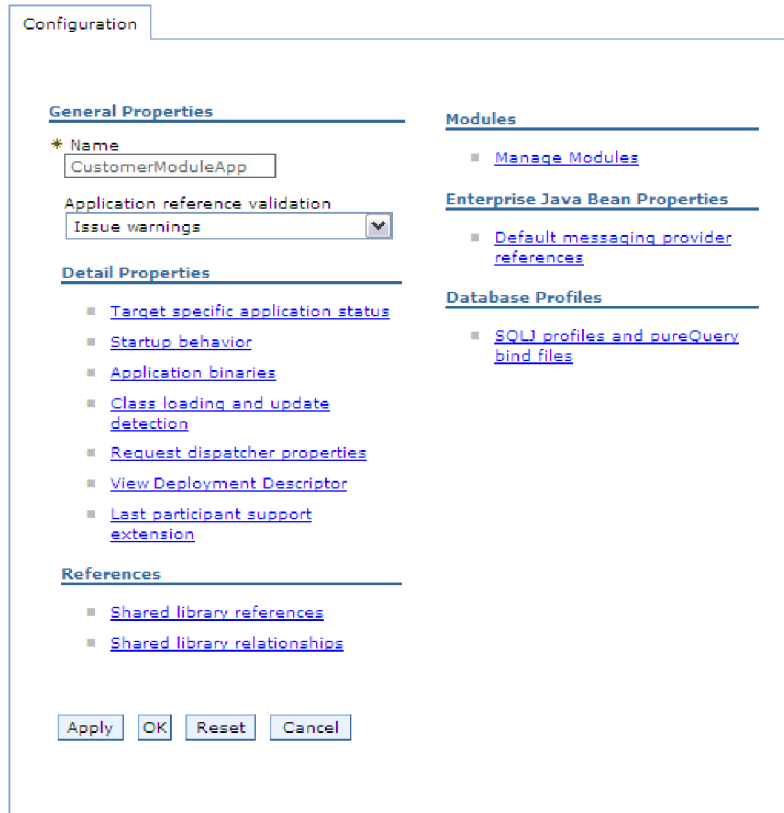


Figure 27. The Manage Modules selection in the Configuration tab

6. Under **Modules**, click **Manage Modules**.
7. Click **IBM WebSphere Adapter for Siebel Business Applications**.
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 168 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.

Related reference

“Activation specification properties” on page 168

Activation specification properties hold the inbound event processing configuration information for a message endpoint. You can set the activation specification properties either through the external service wizard or the administrative console.

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.

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 Siebel Business Applications**.
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 140 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.

Related reference

"Resource adapter properties" on page 140

The resource adapter properties control the general operation of the adapter. You set the resource adapter properties in the external service wizard when you configure the adapter. After deploying the adapter, use the administrative console to change these properties.

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 Siebel application 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 Siebel Business Applications**.
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.

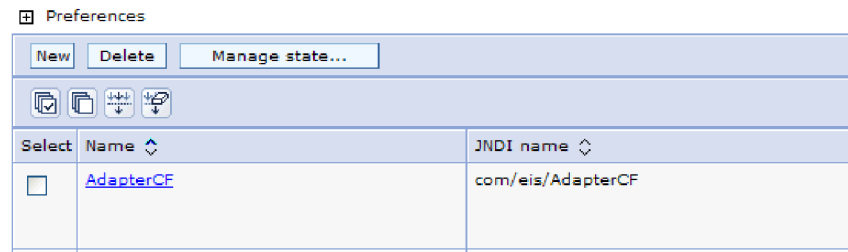


Figure 28. User-defined connection factories for use with the resource adapter

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 Siebel Business Applications. 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 145 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.

Related reference

“Managed connection factory properties” on page 145

Managed connection factory properties are used by the adapter at run time to create an outbound connection instance with the Siebel application.

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 Siebel Business Applications**.
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 168 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.

Related reference

“Activation specification properties” on page 168

Activation specification properties hold the inbound event processing configuration information for a message endpoint. You can set the activation specification properties either through the external service wizard or the administrative console.

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 Siebel Business Applications. 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 110.

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.

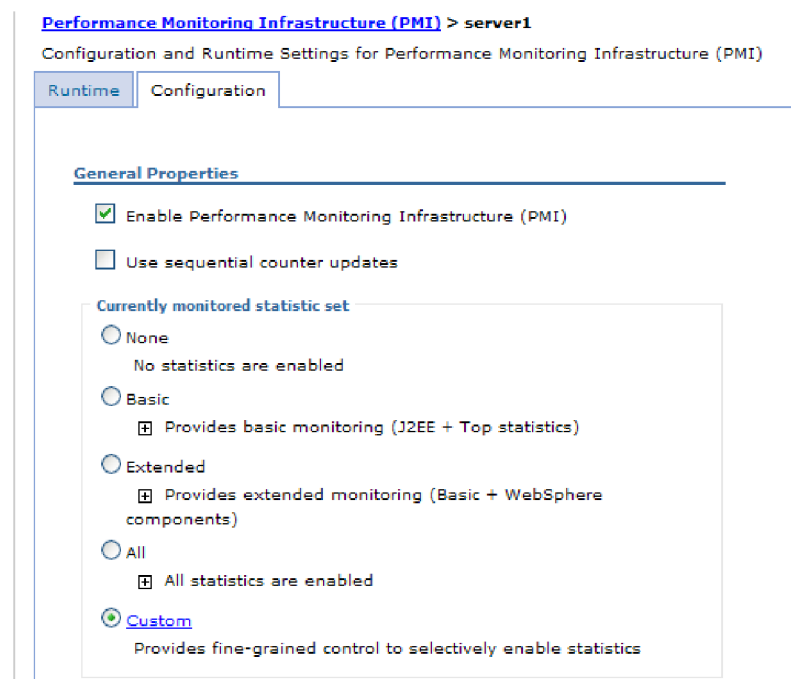


Figure 29. Enabling Performance Monitoring Infrastructure

- 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.

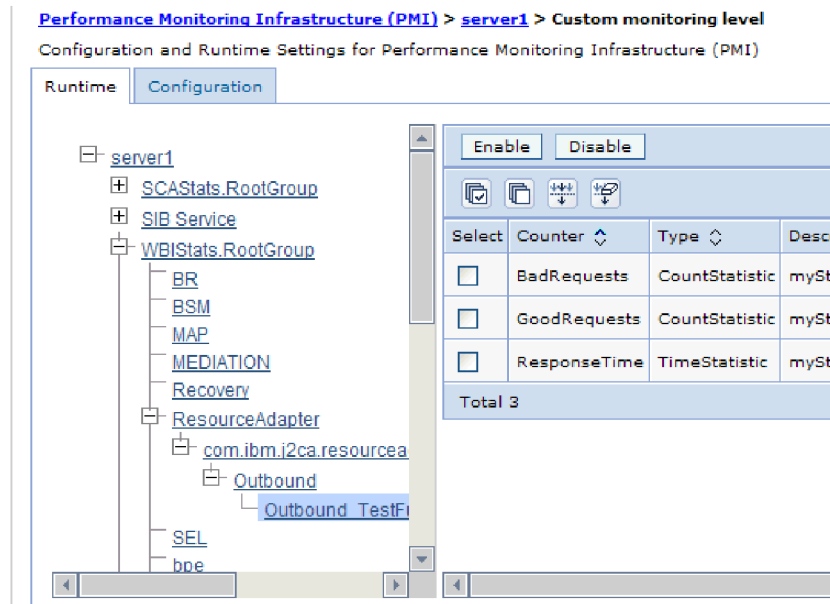


Figure 30. Runtime tab used for configuring PMI

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

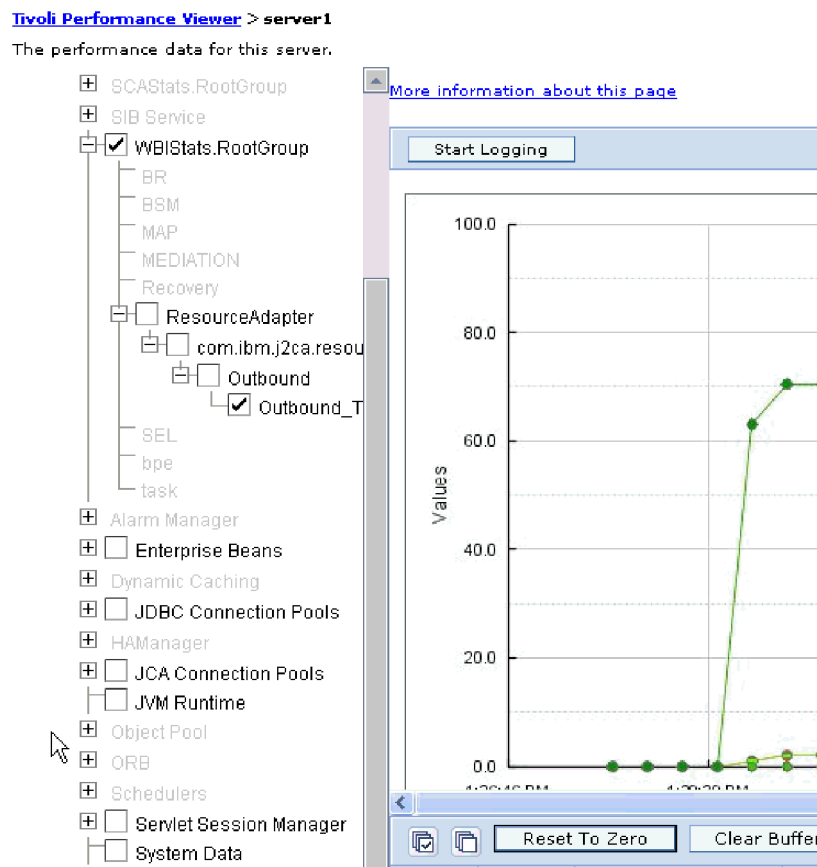


Figure 31. Adapter performance statistics, using graph view

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 external dependencies 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.

Related reference

“Adapter messages” on page 180

View the messages issued by WebSphere Adapter for Siebel Business Applications at the following location.

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/v7r5mx/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 > Application Servers**.
2. Click the name of the server that you want to work with.
3. Under **Troubleshooting**, click **Logs and trace**.
4. Click **Change Log Detail Levels**.
5. Specify when you want the change to take effect:
 - For a static change to the configuration, click the **Configuration** tab.
 - For a dynamic change to the configuration, click the **Runtime** tab.
6. 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 Siebel Business Applications only, select the **com.ibm.j2ca.siebel.*** 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. 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 > 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, in the Modules list, click **Manage Modules**.
4. In the list of modules, click IBM WebSphere Adapter for Siebel Business Applications.
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.

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 thereby 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 located 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.

Solutions to some common problems

Solutions and workarounds to some problems you may encounter while running WebSphere Adapter for Siebel Business Applications with your database are provided. These problems and solutions are also documented as technotes on the Software support website.

For a complete list of technotes about WebSphere Adapters, see <http://www-1.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>.

The adapter may time out based on the default timeout value set on the Siebel application

Problem

The default timeout value for connections set on the Siebel application is approximately 45 minutes. Any connection that is idle (without any transactions coming in) for more than 45 minutes will be timed out. There is no way for the adapter to know this, because the adapter only gets the connection from the container. After it fails, the adapter generates a `CommException` exception back, and the corresponding connection is removed from the pool.

Cause

If the adapter has been idle for more than 45 minutes, it will time out. Also, if the container sends the adapter an invalid connection, the adapter will fail.

Solution and Workaround

If there is a possibility that the adapter remains idle, without any transactions coming in for 60 minutes, increase the timeout value on the Siebel business application to a higher number (for example, 1 1/2 hours). This ensures that the connections remain valid for a longer period. If you cannot increase the timeout value, then resend the failed transaction.

Adapter does not check for the required fields to be set when creating Siebel business components

Problem

When creating Siebel business components, the adapter does not check to ensure that the required fields are set. If required fields are not set, the Siebel enterprise information system (EIS) generates an error, which is returned by the adapter.

Cause

Required fields are not set.

Solution and Workaround

Update the required attributes as needed based on the Siebel exception message provided by the adapter.

Adapter does not throw exception when the call to a Siebel business service method fails

Problem

When the adapter runs a Siebel business service method, it might not throw an exception, even if the transaction fails in the Siebel application.

Cause

This problem occurs because the corresponding business service does not throw an exception but rather returns an empty property set to the adapter. The adapter cannot detect if the transaction was successful, that is, whether the values were updated properly in the Siebel business application. Therefore, the adapter does not throw an exception. Because an empty property set is returned, the adapter returns the corresponding empty top-level business object.

Example scenario: Invalid attribute values are passed through the business object. For example, if the key (for example, "ID") value does not exist and there is no value set on the business object, a corresponding record does not exist in the Siebel business application for that particular key. The business service method might simply ignore the attribute value, even if it is set on the property set. The method must throw an exception but instead returns an empty property set, so the adapter also returns the empty top-level business object (that corresponds to the property set).

Solution and Workaround

Customize the Siebel business service to throw an exception back when failure occurs.

Adapter generates multiple container attributes in the business object for simple link relationships

Problem

Multiple container attributes are generated by IBM WebSphere Adapter for Siebel Business Applications in the parent business object for a simple link relationship between two Siebel business components.

Cause

The adapter generates multiple container attributes for a simple link relationship. For example, as part of the enterprise services discovery generation, assume you select the Siebel business components Account and Business Address in the Siebel business object Account. The corresponding WebSphere business object generated for the Siebel business component Account includes three container attributes, BusinessAddress, BusinessAddress2, and BusinessAddress3, instead of one. You can confirm that SimpleLink application-specific information is set against the three container attributes.

Solution and Workaround

Eventhough the extra container attributes does not affect the transaction if they are left untouched, remove the unwanted attributes using the business object editor in IBM Integration Designer. Therefore, you have only one of the three attributes (BusinessAddress, BusinessAddress2, and BusinessAddress3). For more information on how to use the business object editor, see IBM Integration Designer Information Center.

Adapter does not generate artifacts properly for all the Siebel business components selected

Problem

The adapter does not generate artifacts properly if multiple Siebel business components have the same name after deleting spaces from the names.

Cause

The adapter deletes spaces instead of replacing them with Unicode values, as it does for other special characters.

Solution and Workaround

Use the **Generate business objects with shorter names** option to generate multiple business objects. The naming convention followed for that option adds a counter at the end if resulting names are the same, for example, BusAddr, BusAddr1, BusAddr2.

The adapter does not generate artifacts properly for all selected Siebel business components

Problem

When the adapter generates names for business objects from Siebel business components selected through the external service wizard, it removes special characters, including spaces, from the names. If multiple Siebel business components have the same name after the adapter deletes spaces from the names, the adapter does not generate artifacts properly.

Only one WebSphere business object is generated instead of multiple business objects, even though multiple Siebel business components were selected through the external service wizard.

As an example, consider a Siebel business object called Acc that has two Siebel business components: Bus Addr and BusAddr. Because the adapter deletes spaces from the Siebel business component names to generate the WebSphere business object names, two WebSphere business objects will have the same name: BOAccBCBusAddr. The first WebSphere business object is overwritten by the second one.

Cause

When generating business object names, the adapter deletes spaces instead of replacing them with Unicode values, as it does for other special characters.

Solution and Workaround

Use the **Generate business object with shorter names** option to generate multiple business objects. The naming convention for this option adds a counter to the end of the business object if resulting names will be the same, for example, BusAddr, BusAddr1, and BusAddr2.

The adapter does not generate an exception when the call to a Siebel business service method fails

Problem

When the adapter processes a Siebel business service method and the transaction fails in the Siebel application, the adapter might not generate an exception.

Cause

The corresponding business service does not generate an exception to the adapter when the transaction fails, but instead returns an empty property set to the adapter. Because the adapter cannot detect whether the transaction was successful, that is, whether the values were updated properly in the Siebel application, it does not generate an exception. Because it receives an empty property set, the adapter returns the corresponding empty top-level business object.

For example, if the key value, such as ID, does not exist, and there is no value set on the business object, a corresponding record does not exist in the Siebel application for that particular key. The business service method might ignore the attribute value, even though it is set on the property set. The method must generate an exception but instead returns an empty property set, so the adapter also returns the empty top-level business object that corresponds to the property set.

Solution and Workaround

Customize the Siebel business service to generate an exception when a failure occurs.

Inbound component of WebSphere Adapter for Siebel Business Applications stops working after the number of event entries passes 10,000

Problem

Inbound component of WebSphere Adapter for Siebel Business Applications stops polling and throws some exception.

Cause

WebSphere Adapter for Siebel Business Applications uses Siebel API to access the event table. The possible cause for this issue may be:

- You performed a query or search which is trying to return more rows than is allowed in a result set.
- MaxFetchArraySize or MaxCursorSize parameter values may be set too low.
- Maximum Cursor Size property may be set too low for a specific business component.

Solution and Workaround

Following actions can be considered:

- Modify your query or search specification to return fewer rows in the result set. If not user initiated, check any running workflows, scripts, EAI, charts, or reports that may be executing the query.
- Increase the rows allowed in your result set by modifying the value of MaxFetchArraySize or MaxCursorSize. This will affect the entire Siebel business application. For MaxFetchArraySize, values are:
 - -9,999 (number of rows)
 - 0 (10,000 rows)
 - -1 (unlimited rows)

For instructions on the MaxFetchArraySize parameter, see Siebel Bookshelf version 7.5.3 > Siebel Object Interfaces Reference > Interfaces Reference > Business component Methods > Execute Query. For instructions on the MaxCursorSize parameter, see Siebel Bookshelf version 7.5.3 > Siebel Web Client Administration > Configuration Parameters > Data Source Parameters.

Note: Setting DSMaxFetchArraySize to -1 may cause large memory use, growth of the Siebel Object Manager process, and the Siebel Object Manager to crash from memory exhaustion. For more information, see Alert 951.

- Increase the number of records requested for a specific business component by modifying the values of Maximum Cursor Size property for the specific business component. This will only affect operations performed on the specific business component. For instructions on how to set the Maximum Cursor Size property in Siebel Tools, see Siebel Bookshelf version 7.5.3 > Object Types Reference > Siebel Object Types.

Adapter does not generate WebSphere business objects for child Siebel business components if the related link is inactive

Problem

The Enterprise Metadata Discovery (EMD) portion of the adapter generates WebSphere business objects against the corresponding Siebel business components. If the link that relates the parent business component to the child business component is inactive, the EMD does not generate a WebSphere business object for the corresponding child business component.

Cause

The link in Siebel enterprise information system (EIS) for the parent business component to the child business component is inactive.

Solution and workaround

Set the link as Active in Siebel enterprise information system (EIS).

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 runtime, the adapter will return 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 Siebel Business Applications version 7.5, the following error message is displayed: IBM WebSphere Adapter for Siebel Business Applications has loaded 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. When you install multiple adapters with different CWYBS_AdapterFoundation.jar versions, the adapter returns the ResourceAdapterInternalException message, due to a version conflict. To overcome this conflict, you must migrate all adapters to the same version level. For further assistance, contact WebSphere Adapters Support for help.

Solution

Migrate all adapters to the same version level.

For further assistance, visit http://www-947.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_Adapters_Family.

Endpoint application of the passive adapter instance listens to the events when enableHASupport is set to True

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, version 7.0, 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.

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

Problem

When the EMD is run with the same Siebel RAR file consecutively, and the imported Siebel 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.

Error displayed when using external service discovery wizard

Problem

Error is displayed when using the external service discovery wizard.

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

IBM WebSphere Adapter for Siebel Business Applications returns the following error: 'Failure in connection to EIS. Failed to load the driver: com.siebel.data.SiebelException. Check that the Siebel 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 Siebel driver files again.

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-947.ibm.com/support/entry/portal/Overview/Software/WebSphere/WebSphere_Adapters_Family 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 Siebel Business Applications, 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 Adapters, visit this address:

<http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>.

Plug-in for IBM Support Assistant

WebSphere Adapter for Siebel Business Applications provides a plug-in for IBM Support Assistant, which is a free, local software serviceability workbench. The plug-in supports the dynamic trace feature. For information about installing or using IBM Support Assistant, visit this address:

<http://www.ibm.com/software/support/isa/>.

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 (metadata) about how the adapter must process the business object as well as 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 is metadata that specifies how to process business objects for the adapter.

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 Application-specific information, you can modify the metadata values either from the **Properties** tab in the Java EE perspective of IBM Integration Designer or by using the business object editor.

The business object metadata is generated by the external service wizard at three levels: the business object level (for Siebel business services and Siebel business components), the property level, and the container attribute level.

Metadata at the business object level

Metadata at the business object level defines the top-level wrapper of the business object. The following tables describe the elements within business object metadata for Siebel business objects generated against Siebel business services and Siebel business objects generated against Siebel business components.

Table 8. Metadata elements for business objects generated against Siebel business services

Metadata element	Description
BSN	The name of the business service that is used by the WebSphere Adapter business object.
EventMethod	The event method to use when retrieving event data.
IC	The name of the Siebel integration component that corresponds to the WebSphere Adapter business object.
IO	The name of the Siebel integration object that corresponds to the WebSphere Adapter business object.

Table 9. Metadata elements for business objects generated against Siebel business components

Metadata element	Description
ComponentName	The name of the Siebel business component that corresponds to the WebSphere Adapter business object.
ObjectName	The name of the Siebel business object that corresponds to the WebSphere Adapter business object.

Property-level metadata

Property-level metadata represents either a child object or an array of child objects. The following tables describe the metadata elements of a complex property (a child) or a structure or table property (an array of child objects).

Table 10. Property-level metadata elements generated against Siebel business services

Metadata element	Description
FN=...	Specifies the name of the field in the Siebel business component that corresponds to a WebSphere Adapter business object property.
ParamType	Indicates whether the corresponding field on the Siebel end is input, output, or both.

Table 11. Property-level metadata elements generated against Siebel business components

Metadata element	Description
FieldName	Specifies the name of the field in the Siebel business component that corresponds to a WebSphere Adapter business object property.
PickListKey	Indicates that when a pick list key is specified, a simple attribute will be a PickList.
Restrict=<ParentFieldName>:<childFieldName>	Specifies additional search criteria that can be set to limit the number and type of records that are retrieved.

Container attribute-level metadata

Container attribute-level metadata represents the data located at the container level of business object and components. The following table describes the metadata elements located at the container level for Siebel business objects and Siebel business components.

Table 12. Container attribute-level metadata elements generated against Siebel business objects and Siebel business components

Metadata element	Description
Association	Specifies a 'many to many' relationship at the intersection table when set to true.
FieldName	Identifies the field in the Siebel business component that corresponds to this property.
From=...;To=...	Sets the preprocessing instructions for the adapter to process the To attribute value of the From attribute. The From attribute must be populated, and the To attribute value is valid only if it is null. This is used in Retrieve operations and specifies the child record to be fetched.

Table 12. Container attribute-level metadata elements generated against Siebel business objects and Siebel business components (continued)

Metadata element	Description
KeepRelations	Specifies whether the child objects need to be deleted from the Siebel application. This is only used for Siebel business objects and Siebel business components. It is not used for Siebel business services and Siebel integration objects.
MultiValueLink	Specifies that there is an active multi-value field on the parent object, which indicates an active multi-value link relationship between the parent and child object.
PickList	Indicates a many to one relationship when set to true.
SiebelMessage	Specifies the business service attribute to be used for input and output messages. This is used for Siebel business objects when an attribute represents a Siebel message complex argument. It is valid for both input and output messages, or when one argument is used for the input message and a different one is used for the output message.
SiebelMessageDataType	Specifies whether the attribute is used as the input message, output message, or for both input and output messages. The valid options are Input, Output, and InOut.
SourceField=...;DestinationField=...	Used with simple links to build a simple link relationship between the parent and child WebSphere Adapter business object. The SF, or source field, corresponds to the key attribute (ID) in the WebSphere Adapter parent business object. The DF, or destination field, is the foreign key field in the Siebel child business component. This corresponds to the child attribute with the field name value set to the foreign key field in the Siebel business component.
IOWrapper	Specifies the business service attribute to be used for a wrapper, which contains multiple integration objects. If the business object contains this attribute value, the adapter supports the business services to return different integration objects based on the different input parameters.

Supported operations

An operation is the action that an adapter can perform on the Siebel Business Applications 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 13. Supported operations

Operation	Description
Apply changes	Updates the business component based on the operation that was performed on it. The supported operations are create, update, and delete.
Create	Creates the business component
Delete	Deletes the business component and its child components
Exists	Checks for the existence of incoming business objects
Retrieve	Retrieves the values of the business component
Retrieve all	Retrieves multiple instances of the same business component
Update	Updates the Siebel application with the incoming object

Naming conventions

When the external service wizard generates a business object, it provides a name for the business object based on the name of the object in the Siebel application that it uses to build the business object.

Naming conventions for business objects that represent Siebel business services and integration components

The naming conventions for business objects that represent Siebel business services are the same for both inbound and outbound processing. The names comprise the concatenation of several words, including prefix, business service name, integration object, and method name.

The following table describes the naming conventions that the external service wizard uses to name business objects that represent Siebel business services and integration components.

Table 14. Business object naming conventions for Siebel business services and integration components

Element	Naming convention
Name of the business graph	<Top Level business object Name> + "BG" Note: A "Prefix" is used only for top-level business objects generated against business service methods.
Name of the top-level business object	<Prefix><Business Service Name><Method Name><Names of all the integration objects selected for the Input and InputOutput complex type arguments> <ul style="list-style-type: none"> If there are no Input or Input/Output arguments, the names of all the output arguments are: <Prefix><Business Service Name><Method Name><Names of all the integration objects selected for the output complex type arguments> If the method contains no complex arguments in the method, the naming convention is: <Prefix><Business Service Name><Method Name>
Name of the inbound object generated against integration components	'IO' + <Name of Integration Object> + 'IC' + <Name of integration component> + "BG" The top-level business graph has the suffix BG added to the business object name, as shown in this example: IOAccountInterfaceICAccountBG
Name of the outbound object generated against integration components	'IO' + <Name of Integration Object> + 'IC' + <Name of integration component> The name of an account interface integration object with the integration component account looks like this: IOAccountInterfaceICAccount

Optional: Shorter naming conventions for business objects generated against Siebel business services and integration components

The naming conventions for business objects generated against Siebel business services and integration components are valid if the optional property, 'Generate business objects with shorter names' is specified in the configuration objects window in the external service wizard.

Note: If this optional property is used, you must set the 'Folder' property with a unique value. This avoids overwriting the existing XSDs that were previously generated. For example, if you select 'EAI Siebel Adapter', and click **Query** in two different external service wizard runs for the integration objects, 'Account (PRM ANI)' and 'ATP Check Interface', the top-level object will be named as 'EAISiebelAdapter.xsd'.

The name comprises the concatenation of several words, including prefix, business service name, and integration component name.

The following table describes the naming conventions that the external service wizard uses to name business objects that are generated against Siebel business services and integration components.

Table 15. Shorter business object naming conventions for business objects generated against Siebel business services and integration components

Element	Naming convention
Name of the inbound and outbound child business objects generated against integration components	<Prefix>+<Name of the Siebel Integration Component> Note: The Siebel business object and integration component names are stripped of all non-alphanumeric characters before being added to the business object name. If the resulting names are not unique, a counter is added to the end of the names.
Name of the inbound top-level business object generated against business services and integration components	<Prefix>+<Name of the Siebel Integration Component> + "BG" (with business graph specified) and <Prefix>+<Name of the Siebel Integration Component> (without business graph specified) Note: The Siebel business object and integration component names are stripped of all non-alphanumeric characters before being added to the business object name. If the resulting names are not unique, a counter is added to the end of the names.
Name of the outbound top-level business object generated against business services and integration components	<Prefix>+<Name of the Siebel Business Service> + "BG" (with business graph specified) and <Prefix>+<Name of the Siebel Business Service> (without business graph specified) Note: The Siebel business object and integration component names are stripped of all non-alphanumeric characters before being added to the business object name. If the resulting names are not unique, a counter is added to the end of the names.

Naming conventions for business objects that represent Siebel business objects

The naming conventions for business objects representing Siebel business objects are the same for both inbound and outbound processing. The name comprises the concatenation of several words, including prefix, business object name, and business component name.

The following table describes the naming conventions used by the external service wizard to name business objects that represent Siebel business objects.

Table 16. Business object naming conventions for Siebel business objects

Element	Naming convention
Name of the business object	<Prefix>+<BO>+<Business Object Name>+<BC>+<Business Component Name>. Note: The Siebel business object and component names are stripped of all non-alphanumeric characters before being added to the business object name. If the resulting names are not unique, a counter is added to the end of the names. For example, if two business objects have the name, "SiebelBOAccountBCBusinessAddress", a counter is added as a suffix to make them unique, as shown in this example: "SiebelBOAccountBCAddress1" and "SiebelBOAccountBCAddress2"
Name of the container business object generated for the Exists operation	<SiebelExistsResult> Note: A business graph will not be generated for the "SiebelExistsResult" business object.
Name of the container business object generated for the Retrieve All operation	<Prefix>+BO+<Business Object Name>+<BC>+<Business Component Name>+Container
Name of the top-level business object	<Prefix>+<BO>+<Business Object Name>+<BC>+<Business Component Name>+"BG" Note: The top-level business object will have a business graph generated.

Optional: Shorter naming conventions for business objects generated against Siebel business components

The naming conventions for business objects generated against Siebel business components are valid if the optional property, 'Generate business objects with shorter names' is specified in the configuration objects window in the external service wizard.

Note: If this optional property is used, you must set the 'Folder' property with a unique value. This will avoid overwriting the existing XSDs that were previously generated. For example, Siebel business object -> Siebel business component combination of Account-ESP -> Account and Account (as the top-level object) will be named as, 'Account.xsd'.

The name comprises the concatenation of several words, including prefix and business component name.

The following table describes the naming conventions that the external service wizard uses to name business objects that are generated against Siebel business components.

Table 17. Shorter business object naming conventions for business objects generated against Siebel business components

Element	Naming convention
Name of the top-level business object generated against business components	<p><Prefix>+<Name of the Siebel Business Component> + "BG" (with business graph specified) and <Prefix>+<Name of the Siebel Business Component> (without business graph specified)</p> <p>Note: The Siebel business object and integration component names are stripped of all non-alphanumeric characters before being added to the business object name. If the resulting names are not unique, a counter is added to the end of the names.</p>

Fault business objects

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 wizard creates the following fault business objects:

- DuplicateRecordFault

When processing a Create or an Update 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.
- 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.

Outbound configuration properties

IBM WebSphere Adapter for Siebel Business Applications 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 Siebel Business Applications 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	<p>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.</p> <p>Removing a default value from a required field on the external service wizard <i>will not change that default value</i>. 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.</p> <p>Possible values are Yes and No.</p> <p>Sometimes a property is required only when another property has a specific value. When this is the case, the table will note this dependency. For example,</p> <ul style="list-style-type: none"> • 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	<p>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 will state No default value.</p> <p>The word None is an acceptable default value, and does not mean that there is no default value.</p>
Unit of measure	Specifies how the property is measured, for example in kilobytes or seconds.
Property type	<p>Describes the property type. Valid property types include:</p> <ul style="list-style-type: none"> • Boolean • String • Integer
Usage	<p>Describes usage conditions or restrictions that might apply to the property. For instance, here is how a restriction would be documented:</p> <p>For Rational® Application Developer for WebSphere Software version 6.40 or earlier, the password:</p> <ul style="list-style-type: none"> • Must be uppercase • Must be 8 characters in length <p>For versions of Rational Application Developer for WebSphere Software later than 6.40, the password:</p> <ul style="list-style-type: none"> • Is not case sensitive • Can be up to 40 characters in length. <p>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.</p>

Row	Explanation
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 wizard

The external service wizard properties are used to establish a connection between the external service wizard of IBM Integration Designer, a tool that is used to create business objects, and the Siebel Business Applications server. The properties you configure in the external service wizard specify such things as connection configuration, bidirectional (bidi) properties, and logging and tracing options.

After a connection between the external service wizard and the Siebel Business Applications server is established, the external service wizard is able to access the metadata it needs from the Siebel Business Applications server to create business objects.

Some of the properties that you set in the external service wizard are used as the initial value for resource adapter, managed connection factory, and activation specification properties that you can specify at a later time in the wizard.

The external service wizard 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. For more information about how to read the property detail tables in the sections that follow, see "Guide to information about properties" on page 132.

Note: If you set any of these external service wizard properties using bidirectional script, you must set values that identify the format of the bidirectional script entered for that property.

Table 18. External service wizard properties

Property name in the wizard	Description
Adapter style	Service type associated with the adapter module.
Business object namespace	The namespace value has been added as a prefix to the business object name to keep the business object schemas separated.
Code page	Specifies the code page that the adapter uses to connect to the Siebel Business Applications server.
Connection URL	The connection URL needed to connect to the Siebel application.
Delimiter for keys in the event store	Specifies that the delimiter used between two name value pairs contains the object key name and value.

Table 18. External service wizard properties (continued)

Property name in the wizard	Description
Folder	The location of the generated business object.
Generate business objects with shorter names	Ensures that the adapter generates shorter business object names, which are based on the Siebel integration components, business services, and business components rather than the concatenation of several words (which is the default).
Language code	The language code used to log on to the Siebel Business Applications server.
Method name	The name of the business service method to be implemented.
Password	The password for the corresponding user name.
Prefix for business object names	The prefix for the business object name.
Siebel business object name for event store	The name of the business object in the event store where events are stored for inbound processing.
Siebel repository name	The name of the Siebel repository from which the objects are to be discovered.
Siebel server view mode	Specifies the Siebel Business Applications mode and controls the data type to be retrieved and what actions can be performed.
Type of Siebel objects to discover	The type of Siebel objects (business objects or business services) that need to be discovered and listed.
Use resonate support for load balancing on Siebel server	Specifies that if resonate support is installed on the Siebel Business Applications server, and the value is set to true, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
User name	The user name that is used to log in to the Siebel application.

Adapter style (AdapterStyle)

This property specifies the service type associated with the adapter module.

Table 19. Service type details

Required	Yes
Default	Outbound
Property type	List of values
Possible values	Outbound Inbound
Usage	Specifies the service type associated with the adapter module
Globalized	No
Bidi supported	No

Business object namespace (BusinessObjectNameSpace)

This property specifies that the namespace value has been added as a prefix to the business object name to keep the business object schemas separated.

Table 20. Business object namespace details

Required	Yes
Default	http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel
Property type	String
Usage	The namespace value is added as a prefix to the business object name to keep the business object schemas separated.
Example	http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount
Globalized	No
Bidi supported	No

Connection URL (ConnectionURL)

This property specifies the connection URL needed to connect to the Siebel application.

Table 21. Connection URL details

Required	Yes
Default	No default value
Property type	String
Usage	The connection URLs for all versions of Siebel follow this format: Protocol://machinename:port/enterprisename/object manager/server name. . The default port number is 2320. For Siebel version 7.5x and earlier versions, the port number (2320) and server name are specified, as shown in the example below. For Siebel version 7.8, the port and server name are not specified, as shown in the example below. If you do not select the default port, then you can specify another port number (for example, 2321), as shown in the example below for Siebel version 8.
Examples	The following sample connection URLs are for different versions of Siebel: <ul style="list-style-type: none"> • For Siebel 7.5: siebel://<IP_address>:2320/siebel/SSEObjMgr_ENU/sebldev1 . • For Siebel 7.8: siebel://<IP_address>/Sieb78/SSEObjMgr_enu . • For Siebel 8: siebel://<IP_address>:2321/SBA_80/SSEObjMgr_enu .
Globalized	Yes
Bidi supported	Yes

Code page (CodePage)

Specifies the code page that the adapter uses to connect to Siebel Business Applications server.

Table 22. Code page details

Required	No
Default	No default value
Property type	String
Usage	The Siebel Java Data Bean supports a limited list of code page settings. If the current code page setting of your tooling is not supported, the adapter may not be able to connect to the Siebel Business Applications server. This property allows the users to specify the code page details that the adapter uses to connect to Siebel Business Applications server for metadata discovery.
Globalized	No

Table 22. Code page details (continued)

Bidi supported	No
----------------	----

Delimiter for keys in the event store (DelimiterForKeysInTheEventStore)

Table 23. Delimiter for keys in the event store details

Required	Yes
Default	;
Property type	String
Usage	This is the delimiter used between two name value pairs containing the object key name and value.
Examples	The default value for this property can be changed, however, if you remove the default value and do not set it again, the default value (;) will be used. If the event table key field has values, such as AccountId=1-314:Id=1-325, then the event delimiter is the colon (:). The object key names are AccountId and Id. The values are 1-314 and 1-325.
Globalized	Yes
Bidi supported	Yes

Folder (Folder)

This property specifies the location of the generated business objects.

Table 24. Folder details

Required	No
Default	No default value
Property type	String
Usage	The generated business objects are copied into this folder.
Example	inboundartifacts and outboundartifacts
Globalized	No
Bidi supported	No

Generate business objects with shorter names (GenerateBusinessObjectsWithShorterNames)

This property ensures that the adapter generates shorter business object names, which are based on the Siebel integration components, business services, and business components rather than the concatenation of several words (which is the default).

Table 25. Generate business objects with shorter names details

Required	No
Default	No default value
Property type	Boolean

Table 25. Generate business objects with shorter names details (continued)

Usage	Ensures that the adapter generates shorter business object names. The shorter business object names are based on the Siebel integration components, business services, and business components. The prefix is also attached to the shorter names. Note: The adapter removes special characters from the shorter business object names. Alphanumeric characters (a-z, A-Z, and 1-9) are supported, and a counter (1-9) is added to the end of business object names to avoid duplication of names.
Example	If 'Account' is the name of the Siebel business component, and 'Siebel' is the prefix, the shorter name, is 'Siebel_Account'.
Globalized	No
Bidi supported	No

Language code (LanguageCode)

This property specifies the language code used to log on to the Siebel Business Applications server.

Table 26. Language code details

Required	Yes
Default	ENU
Property type	String
Usage	If the system locale is English, the value for this property is ENU (English). This is used to log on to the Siebel Business Applications server.
Globalized	No
Bidi supported	No

Method name (MethodName)

This property specifies the name of the business service method to be implemented.

Table 27. Method name details

Required	Yes
Default	Query
Property type	String
Usage	The default is Query.
Example	Query, QueryByExample, QueryById, and so on.
Globalized	Yes
Bidi supported	Yes

Password (Password)

This property specifies the password for the corresponding user name.

Table 28. Password details

Required	Yes
Default	No default value
Property type	String

Table 28. Password details (continued)

Usage	If a J2C Authentication Alias is used, then a password is not required.
Example	1-XYZ
Globalized	Yes
Bidi supported	Yes

Prefix for business object names (PrefixForBusinessObjectNames)

This property specifies the prefix for the business object name.

Table 29. Prefix details

Required	No
Default	No default value
Property type	String
Usage	The prefix string is attached to the front of the business object name that was generated.
Example	You use the prefix, IBM and generate a business object for the EAI Siebel Adapter and the Insert method. Then, you choose the Account Interface and Business Address Interface integration object against an Input/Output method argument. The corresponding business object generated would be: IBMEAISiebelAdapterInsertAccountInterfacBusinessAddressInterface.
Globalized	Yes
Bidi supported	Yes

Siebel business object name for event store (SiebelBusinessObjectNameForEventStore)

This property specifies the name of the business object in the event store where events are stored for inbound processing.

Table 30. Siebel business object name for event store details

Required	Yes
Default	IBM_EVENT
Property type	String
Usage	After clicking the advanced button on the connection properties window on the external service wizard, this property will display under the "Event configuration" tab. The two values listed are IBM_EVENT and IBM2. If you create a custom event component name, you can specify the value for it in the list box.
Globalized	Yes
Bidi supported	No

Siebel repository name (SiebelRepositoryName)

This property specifies the name of the Siebel repository from which the objects are to be discovered.

Table 31. Siebel repository name details

Required	Yes
Default	Siebel Repository

Table 31. Siebel repository name details (continued)

Property type	String
Usage	This default value is Siebel Repository. Although this is a required field, it is optional on the external service wizard. You can edit this value to point to other repositories if needed.
Globalized	No
Bidi supported	No

Siebel server view mode (SiebelServerViewMode)

This property specifies the Siebel Business Applications server view mode and controls the data that can be retrieved and what actions can be performed on it.

Table 32. Siebel server view mode details

Required	Yes
Default	3
Property type	Integer
Usage	This property displays after clicking the advanced button on the connection properties window on the external service wizard. This mode, when set to "Type of Siebel objects to discover" applies only to Siebel business objects, not to Siebel business services. The values supported by Siebel are 1 to 9.
Globalized	No

Type of Siebel objects to discover (TypeOfSiebelObjectsToDiscover)

This property specifies the type of Siebel object that needs to be discovered and listed.

Table 33. Type of Siebel objects to discover details

Required	Yes
Possible values	Siebel business objects or Siebel business services
Default	Siebel business objects
Property type	String
Usage	Although the default is Siebel business objects, you can select Siebel business services. Based on your selection, the external service wizard retrieves either the business objects or the business services.
Globalized	No
Bidi supported	No

Use resonate support for load balancing on Siebel server (UseResonateSupportForLoadBalancingOnSiebelServer)

This property indicates whether the Siebel Business Applications server uses resonate support.

Table 34. Use resonate support for load balancing on Siebel server details

Required	No
Possible values	True False

Table 34. Use resonate support for load balancing on Siebel server details (continued)

Default	True
Property type	Boolean
Usage	This property displays after clicking the advanced button on the connection properties window on the external service wizard. If the check box is selected, it will be set to True, and the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently. If the check box is not selected, it will be set to false.
Globalized	No

User name (UserName)

This property specifies the user name that is used to log in to the Siebel application.

Table 35. User name details

Required	Yes
Default	No default value
Property type	String
Usage	If a J2C Authentication Alias is used, then a user name is not required.
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 in 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 no longer required in version 7.0, but are supported for compatibility with previous versions:

- LogFileMaxSize
- LogFileName
- LogNumberOfFiles
- TraceFileMaxSize
- TraceFileName
- TraceNumberOfFiles

The following table lists and describes the resource adapter properties. A more detailed description of each property is provided in the sections that follow the table. For more information about how to read the property detail tables in the sections to follow, see “Guide to information about properties” on page 132.

Table 36. Resource adapter properties

Property name		Description
In the wizard	In the administrative console	
Adapter ID	AdapterID	Identifies the adapter instance for PMI events and for logging and tracing.

Table 36. Resource adapter properties (continued)

Property name		Description
In the wizard	In the administrative console	
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.
Event delimiter	EventDelimiter	Specifies whether the delimiter used between two name value pairs contains the object key name and value.
(Not available)	LogFileMaxSize	Deprecated
(Not available)	LogFilename	Deprecated
(Not available)	LogNumberOfFiles	Deprecated
Resonate support	resonateSupport	Specifies that if resonate support is installed on the Siebel Business Applications server, and the value is set to True, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
Siebel server view mode	viewMode	Specifies the Siebel view mode and controls the type of data to be retrieved and the actions to be performed.
(Not available)	TraceFileMaxSize	Deprecated
(Not available)	TraceFileName	Deprecated
(Not available)	TraceNumberOfFiles	Deprecated

Adapter ID (AdapterID)

This property identifies a specific deployment or instance of the adapter.

Table 37. Adapter ID details

Required	Yes
Default	001
Property type	String

Table 37. Adapter ID details (continued)

Usage	<p>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, SEBLRA, 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 SEBLRA001.</p> <p>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.</p> <p>For example, when you set the adapter ID property of two instances of WebSphere Adapter for Siebel Business Applications to 001 and 002. The component IDs for those instances, SEBLRA001 and SEBLRA002, 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 SEBLRAInstanc.</p> <p>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.</p>
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 38. Disguise user data as "XXX" in log and trace files details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	<p>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.</p> <p>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.</p>
Globalized	No

Table 38. Disguise user data as "XXX" in log and trace files details (continued)

Bidi supported	No
----------------	----

Enable high availability support (enableHASupport)

Do not change this property. It must be set to true.

Event delimiter (EventDelimiter)

This property indicates that the delimiter used between two name value pairs contains the object key name and value.

Table 39. Event delimiter details

Required	Yes
Default	;
Property type	String
Usage	If multiple value pairs are set against the object key in the event component, they will be used for the delimiter.
Globalized	No

Log file maximum size (LogFileMaxSize)

This property specifies the size of the log files in kilobytes.

Table 40. 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 41. 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 42. 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

Resonate support (ResonateSupport)

This property indicates whether the Siebel Business Applications server uses resonate support.

Table 43. Resonate support details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If the value of this property is set to True, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
Globalized	No

Siebel server view mode (SiebelServerViewMode)

This property specifies the Siebel view mode and controls the data that can be retrieved and the actions to be performed on it.

Table 44. View mode details

Required	Yes
Default	3
Property type	Integer
Usage	The View mode property applies only to Siebel business objects and not to Siebel business services.
Globalized	No

Trace file maximum size (TraceFileMaxSize)

This property specifies the size of the trace files in kilobytes.

Table 45. Trace file maximum size details

Required	No
Default	0

Table 45. Trace file maximum size details (continued)

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 46. 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 47. Trace number of files details

Required	No
Default	1
Property type	Integer
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 instance with the Siebel application.

You set the managed connection factory properties through either the external service wizard or the administrative console (after deployment).

The following table lists the managed connection factory properties for inbound communication. You set the activation specification properties using the external service wizard and can change them using the Integration Designer Assembly Editor, or after deployment through the administrative console. 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 understanding property details.

Note: The external service wizard refers to these properties as managed connection factory properties and the IBM Business Process Manager or WebSphere Enterprise Service Bus administrative console refers to them as (J2C) connection factory properties.

Table 48. Managed connection factory properties

Property name		Description
In the wizard	In the administrative console	
Adapter ID	AdapterID	Identifies the adapter instance for PMI events and for logging and tracing.
Connection URL	connectString	The connection URL needed to connect to the Siebel application.
Code page	codePage	Specifies the code page that the adapter uses to connect to Siebel server.
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.
Include empty elements in the output	includeEmptyField	Specifies if the empty elements are included in the response business object.
Language code	languageCode	The language code used to log on to the Siebel Business Applications server.
"Maximum retries in case of system connection failure (connectionRetryLimit)" on page 150	connectionRetryLimit	Specifies the maximum number of times the adapter tries to reestablish an outbound connection to the EIS.
Password	password	The password for the corresponding user name.
Prefix	prefix	The prefix for the business object name.
"Retry interval if connection fails (in milliseconds) (connectionRetryInterval)" on page 151	connectionRetryInterval	Specifies the time interval between attempts to reconnect to the Siebel Business Applications server if the connection fails.
Resonate support	resonateSupport	Specifies that if resonate support is installed on the Siebel Business Applications server, and the value is set to True, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
"Show error on empty result set (errorOnEmptyResultSet)" on page 152	errorOnEmptyResultSet	For a business object, the property specifies if the adapter returns an empty business object or RecordNotFoundException when the result of RetrieveAll operation does not return any records. For a business service, the property specifies if the adapter generates an exception when a returned Siebel message is null.
User name	userName	The user name that is used to log in to the Siebel application
View mode	viewMode	Specifies the Siebel view mode and controls the data that can be retrieved and what actions can be performed on it.

Adapter ID (AdapterID)

This property identifies a specific deployment or instance of the adapter.

Table 49. Adapter ID details

Required	Yes
Default	001
Property type	String
Usage	<p>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, SEBLRA, 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 SEBLRA001.</p> <p>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.</p> <p>For example, when you set the adapter ID property of two instances of WebSphere Adapter for Siebel Business Applications to 001 and 002. The component IDs for those instances, SEBLRA001 and SEBLRA002, 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 SEBLRAInstanc.</p> <p>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.</p>
Globalized	Yes
Bidi supported	No

Code page (codePage)

Specifies the code page that the adapter uses to connect to Siebel server.

Table 50. Code page details

Required	No
Default	No default value
Property type	String

Table 50. Code page details (continued)

Usage	<p>The Siebel Java Data Bean supports a limited list of code page settings. If the current code page setting of your run time is not supported, the adapter may not be able to connect to the Siebel server.</p> <p>This property allows the users to specify the code page details that the adapter uses to connect to Siebel server.</p> <p>Note: The adapter uses the code page property to change the “file.encoding” property of the current Java Virtual Machine (JVM), when it establishes its first connection to the Siebel server. This may affect other applications which run on the same JVM. You need to ensure that the code page has a valid value for all applications.</p>
Globalized	No
Bidi supported	No

Connection URL (ConnectionURL)

This property specifies the connection URL needed to connect to the Siebel application.

Table 51. Connection URL details

Required	Yes
Default	No default value
Property type	String
Usage	<p>Protocol://machinename:port/enterprisename/object manager/server name.</p> <ul style="list-style-type: none"> • For Siebel 7.0.5 to 7.5x: siebel://<IP ADDRESS>/siebel/SSEObjMgr_ENU/seb1dev1. • For Siebel 7.8: siebel://<IP ADDRESS>:2321/Sieb78/SSEObjMgr_enu. • For Siebel 8: siebel://<IP ADDRESS>:2321/SBA_80/SSEObjMgr_enu. The default port number is 2320. <p>In the examples (for Siebel versions 7.8 and 8), another port (2321) has been specified.</p>
Globalized	Yes
Bidi supported	Yes

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 52. Disguise user data as "XXX" in log and trace files details

Required	No
Possible values	True False
Default	False
Property type	Boolean

Table 52. Disguise user data as "XXX" in log and trace files details (continued)

Usage	<p>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.</p> <p>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.</p>
Globalized	No
Bidi supported	No

Include empty elements in the output (includeEmptyField)

Specifies if the adapter returns the elements that have an empty string value in the response business object.

Table 53. Include empty elements in the output details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	This property enables you to include the empty elements in the output. When this property is selected, the adapter returns the elements that have empty string values in the response business object.
Globalized	No
Bidi supported	No

Language code (LanguageCode)

This property specifies the language code used to log on to the Siebel Business Applications server.

Table 54. Language code details

Required	Yes
Possible values	None
Default	ENU
Property type	String
Usage	If the system locale is English, the value for this property is ENU (English). It is used to log on to the Siebel Business Applications server.
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 Siebel Business Applications server, when the adapter encounters an error related to the outbound connection.

Table 55. 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
Usage	<p>When this property is set to:</p> <p>0</p> <ul style="list-style-type: none"> The adapter does not attempt to reconnect to the Siebel Business Applications server, if an error occurs during startup or while establishing a connection. The adapter does not verify if the connection to the Siebel Business Applications server is valid when there is an outbound request during run time. <p>>0</p> <ul style="list-style-type: none"> The adapter attempts to reconnect to the Siebel Business Applications 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 Siebel Business Applications 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. <p>If the adapter fails to establish a connection after trying for the specified number of times, a connection error is generated.</p> <p>If the adapter is successful in reestablishing the connection, the outbound operation is completed.</p>
Globalized	No
Bidi supported	No

Password (Password)

This property specifies the password for the corresponding user name.

Table 56. Password details

Required	Yes
Default	No default value
Property type	String
Example	sadmin
Usage	<p>This property will display after clicking the advanced button on the connection properties window on the external service wizard. The password is saved in .import and .export files. This makes it possible for the adapter to connect to the Siebel application after it has been deployed. If a J2C Authentication Alias is used, then a password is not required.</p>
Globalized	Yes

Table 56. Password details (continued)

Bidi supported	Yes
----------------	-----

Prefix (Prefix)

This property specifies the prefix for the business object name.

Table 57. Prefix details

Required	No
Default	No default value
Property type	String
Usage	The prefix string is attached to the front of the business object name.
Example	Use the prefix, IBM and generate a business object for the EAI Siebel Adapter and the Insert method. Then you choose the integration object, Account (PRM ANI). The corresponding business object generated is: IBMEAISiebelAdapterInsertAccountU40PRMANIU41, where U40 and U41 are the unicode value replacements of (and).
Globalized	Yes
Bidi supported	Yes

Retry interval if connection fails (in milliseconds) (connectionRetryInterval)

This property specifies the time interval between attempts to reconnect to the Siebel Business Applications server if the connection fails.

Table 58. 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 is set to greater than 0. When the adapter encounters an error while establishing a connection to the Siebel Business Applications server, this property specifies the time interval the adapter waits between attempts to reestablish a connection.
Globalized	No
Bidi supported	No

Resonate support (ResonateSupport)

This property indicates whether the Siebel Business Applications server uses resonate support.

Table 59. Resonate support details

Required	No
Possible values	True False

Table 59. Resonate support details (continued)

Default	True
Property type	Boolean
Usage	If the property is selected (value set to True), and the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
Globalized	No

Show error on empty result set (errorOnEmptyResultSet)

For a business object, this property specifies if the adapter returns an empty business object or RecordNotFoundException when the result of RetrieveAll operation does not return any records. For a business service, this property specifies if the adapter generates an exception when a returned Siebel message is null.

Table 60. Show error on empty result set details

Required	No
Default	For business object: True For business service: False
Property type	Boolean
Usage	For business object, when this property is set to: <ul style="list-style-type: none"> • True: Throws a RecordNotFoundException if no records are found in RetrieveAll operation. • False: Returns an empty business object if no records are found in RetrieveAll operation. For business service, when this property is set to: <ul style="list-style-type: none"> • True: Throws a RecordNotFoundException if no records are returned. • False: Returns null Siebel message if no records are returned.
Globalized	No
Bidi supported	No

User name (UserName)

This property specifies the user name that is used to log in to the Siebel application.

Table 61. User name details

Required	Yes
Possible values	None
Default	No default value
Property type	String
Usage	This property will display after clicking the advanced button on the connection properties window on the external service wizard. The user name is saved in .import and .export files. This makes it possible for the adapter to connect to the Siebel application after it has been deployed. If a J2C Authentication Alias is used, then a password is not required.

Table 61. User name details (continued)

Globalized	Yes
Bidi supported	Yes

View mode (ViewMode)

This property specifies the Siebel view mode and controls the data that can be retrieved and what actions can be performed on it.

Table 62. View mode details

Required	Yes
Possible values	1 - 9
Default	3
Property type	Integer
Usage	The View mode property applies only to Siebel business objects and not to Siebel business services. When this property is used for Siebel business objects, the default is 3.
Examples	The adapter supports values 1 - 9. For example, 1 is Manager View, 2 is Personal View, and 3 is All View.
Globalized	No

Related tasks

“Setting deployment properties and generating the service ” on page 56

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 Siebel Business Applications. Then, the wizard creates a business integration module where all the artifacts and property values are saved.

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 assembly editor in IBM Integration Designer. The properties reside in the method binding of the import.

Table 63 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 132.

Table 63. Interaction specification property for the WebSphere Adapter for Siebel Business Applications

Property name		Description
In the wizard	In the assembly editor	
“Maximum records (MaximumRecords)” on page 154	MaximumRecords	Maximum number of records to return during a RetrieveAll operation

Maximum records (MaximumRecords)

This property specifies the maximum number of records to return during a RetrieveAll operation.

Table 64. Maximum records details

Required	Yes
Default	100
Usage	If the number of hits in the database exceeds the value of the MaximumRecords property, the adapter returns the MatchesExceededLimitException error. The adapter uses this property to help avoid out-of-memory issues.
Property type	Integer
Globalized	No
Bidi supported	No

Related tasks

Chapter 6, “Changing interaction specification properties,” on page 83
To change interaction specification properties for your adapter module after generating the service, use the assembly editor in IBM Integration Designer.

Inbound configuration properties

WebSphere Adapter for Siebel Business Applications 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 Siebel Business Applications 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	<p>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.</p> <p>Removing a default value from a required field on the external service wizard <i>will not change that default value</i>. 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.</p> <p>Possible values are Yes and No.</p> <p>Sometimes a property is required only when another property has a specific value. When this is the case, the table will note this dependency. For example,</p> <ul style="list-style-type: none"> • 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	<p>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 will state No default value.</p> <p>The word None is an acceptable default value, and does not mean that there is no default value.</p>
Unit of measure	Specifies how the property is measured, for example in kilobytes or seconds.
Property type	<p>Describes the property type. Valid property types include:</p> <ul style="list-style-type: none"> • Boolean • String • Integer
Usage	<p>Describes usage conditions or restrictions that might apply to the property. For instance, here is how a restriction would be documented:</p> <p>For Rational Application Developer for WebSphere Software version 6.40 or earlier, the password:</p> <ul style="list-style-type: none"> • Must be uppercase • Must be 8 characters in length <p>For versions of Rational Application Developer for WebSphere Software later than 6.40, the password:</p> <ul style="list-style-type: none"> • Is not case sensitive • Can be up to 40 characters in length. <p>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.</p>
Example	<p>Provides sample property values, for example:</p> <p>"If Language is set to JA (Japanese), code page number is set to 8000".</p>
Globalized	<p>If a property is globalized, it has national language support, meaning that you can set the value in your national language.</p> <p>Valid values are Yes and No.</p>

Row	Explanation
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 wizard

The external service wizard properties are used to establish a connection between the external service wizard of IBM Integration Designer, a tool that is used to create business objects, and the Siebel Business Applications server. The properties you configure in the external service wizard specify such things as connection configuration, bidirectional (bidi) properties, and logging and tracing options.

After a connection between the external service wizard and the Siebel Business Applications server is established, the external service wizard is able to access the metadata it needs from the Siebel Business Applications server to create business objects.

Some of the properties that you set in the external service wizard are used as the initial value for resource adapter, managed connection factory, and activation specification properties that you can specify at a later time in the wizard.

The external service wizard 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. For more information about how to read the property detail tables in the sections that follow, see “Guide to information about properties” on page 132.

Note: If you set any of these external service wizard properties using bidirectional script, you must set values that identify the format of the bidirectional script entered for that property.

Table 65. External service wizard properties

Property name in the wizard	Description
Adapter style	Service type associated with the adapter module.
Business object namespace	The namespace value has been added as a prefix to the business object name to keep the business object schemas separated.
Code page	Specifies the code page that the adapter uses to connect to the Siebel Business Applications server.
Connection URL	The connection URL needed to connect to the Siebel application.
Delimiter for keys in the event store	Specifies that the delimiter used between two name value pairs contains the object key name and value.
Folder	The location of the generated business object.
Generate business objects with shorter names	Ensures that the adapter generates shorter business object names, which are based on the Siebel integration components, business services, and business components rather than the concatenation of several words (which is the default).

Table 65. External service wizard properties (continued)

Property name in the wizard	Description
Language code	The language code used to log on to the Siebel Business Applications server.
Method name	The name of the business service method to be implemented.
Password	The password for the corresponding user name.
Prefix for business object names	The prefix for the business object name.
Siebel business object name for event store	The name of the business object in the event store where events are stored for inbound processing.
Siebel repository name	The name of the Siebel repository from which the objects are to be discovered.
Siebel server view mode	Specifies the Siebel Business Applications mode and controls the data type to be retrieved and what actions can be performed.
Type of Siebel objects to discover	The type of Siebel objects (business objects or business services) that need to be discovered and listed.
Use resonate support for load balancing on Siebel server	Specifies that if resonate support is installed on the Siebel Business Applications server, and the value is set to true, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
User name	The user name that is used to log in to the Siebel application.

Adapter style (AdapterStyle)

This property specifies the service type associated with the adapter module.

Table 66. Service type details

Required	Yes
Default	Outbound
Property type	List of values
Possible values	Outbound Inbound
Usage	Specifies the service type associated with the adapter module
Globalized	No
Bidi supported	No

Business object namespace (BusinessObjectNameSpace)

This property specifies that the namespace value has been added as a prefix to the business object name to keep the business object schemas separated.

Table 67. Business object namespace details

Required	Yes
Default	http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel
Property type	String

Table 67. Business object namespace details (continued)

Usage	The namespace value is added as a prefix to the business object name to keep the business object schemas separated.
Example	http://www.ibm.com/xmlns/prod/wbi/j2ca/siebel/IBMSiebelAccountInsertAccount
Globalized	No
Bidi supported	No

Connection URL (ConnectionURL)

This property specifies the connection URL needed to connect to the Siebel application.

Table 68. Connection URL details

Required	Yes
Default	No default value
Property type	String
Usage	The connection URLs for all versions of Siebel follow this format: Protocol://machinename:port/enterprisename/object manager/server name. . The default port number is 2320. For Siebel version 7.5x and earlier versions, the port number (2320) and server name are specified, as shown in the example below. For Siebel version 7.8, the port and server name are not specified, as shown in the example below. If you do not select the default port, then you can specify another port number (for example, 2321), as shown in the example below for Siebel version 8.
Examples	The following sample connection URLs are for different versions of Siebel: <ul style="list-style-type: none"> • For Siebel 7.5: siebel://<IP_address>:2320/siebel/SSEObjMgr_ENU/sebldev1 . • For Siebel 7.8: siebel://<IP_address>/Sieb78/SSEObjMgr_enu . • For Siebel 8: siebel://<IP_address>:2321/SBA_80/SSEObjMgr_enu .
Globalized	Yes
Bidi supported	Yes

Code page (CodePage)

Specifies the code page that the adapter uses to connect to Siebel Business Applications server.

Table 69. Code page details

Required	No
Default	No default value
Property type	String
Usage	The Siebel Java Data Bean supports a limited list of code page settings. If the current code page setting of your tooling is not supported, the adapter may not be able to connect to the Siebel Business Applications server. This property allows the users to specify the code page details that the adapter uses to connect to Siebel Business Applications server for metadata discovery.
Globalized	No
Bidi supported	No

Delimiter for keys in the event store (DelimiterForKeysInTheEventStore)

Table 70. Delimiter for keys in the event store details

Required	Yes
Default	;
Property type	String
Usage	This is the delimiter used between two name value pairs containing the object key name and value.
Examples	The default value for this property can be changed, however, if you remove the default value and do not set it again, the default value (;) will be used. If the event table key field has values, such as AccountId=1-314:Id=1-325, then the event delimiter is the colon (:). The object key names are AccountId and Id. The values are 1-314 and 1-325.
Globalized	Yes
Bidi supported	Yes

Folder (Folder)

This property specifies the location of the generated business objects.

Table 71. Folder details

Required	No
Default	No default value
Property type	String
Usage	The generated business objects are copied into this folder.
Example	inboundartifacts and outboundartifacts
Globalized	No
Bidi supported	No

Generate business objects with shorter names (GenerateBusinessObjectsWithShorterNames)

This property ensures that the adapter generates shorter business object names, which are based on the Siebel integration components, business services, and business components rather than the concatenation of several words (which is the default).

Table 72. Generate business objects with shorter names details

Required	No
Default	No default value
Property type	Boolean
Usage	Ensures that the adapter generates shorter business object names. The shorter business object names are based on the Siebel integration components, business services, and business components. The prefix is also attached to the shorter names. Note: The adapter removes special characters from the shorter business object names. Alphanumeric characters (a-z, A-Z, and 1-9) are supported, and a counter (1-9) is added to the end of business object names to avoid duplication of names.
Example	If 'Account' is the name of the Siebel business component, and 'Siebel' is the prefix, the shorter name, is 'Siebel_Account'.

Table 72. Generate business objects with shorter names details (continued)

Globalized	No
Bidi supported	No

Language code (LanguageCode)

This property specifies the language code used to log on to the Siebel Business Applications server.

Table 73. Language code details

Required	Yes
Default	ENU
Property type	String
Usage	If the system locale is English, the value for this property is ENU (English). This is used to log on to the Siebel Business Applications server.
Globalized	No
Bidi supported	No

Method name (MethodName)

This property specifies the name of the business service method to be implemented.

Table 74. Method name details

Required	Yes
Default	Query
Property type	String
Usage	The default is Query.
Example	Query, QueryByExample, QueryById, and so on.
Globalized	Yes
Bidi supported	Yes

Password (Password)

This property specifies the password for the corresponding user name.

Table 75. Password details

Required	Yes
Default	No default value
Property type	String
Usage	If a J2C Authentication Alias is used, then a password is not required.
Example	1-XYZ
Globalized	Yes
Bidi supported	Yes

Prefix for business object names (PrefixForBusinessObjectNames)

This property specifies the prefix for the business object name.

Table 76. Prefix details

Required	No
Default	No default value
Property type	String
Usage	The prefix string is attached to the front of the business object name that was generated.
Example	You use the prefix, IBM and generate a business object for the EAI Siebel Adapter and the Insert method. Then, you choose the Account Interface and Business Address Interface integration object against an Input/Output method argument. The corresponding business object generated would be: IBMEAISiebelAdapterInsertAccountInterfacBusinessAddressInterface.
Globalized	Yes
Bidi supported	Yes

Siebel business object name for event store (SiebelBusinessObjectNameForEventStore)

This property specifies the name of the business object in the event store where events are stored for inbound processing.

Table 77. Siebel business object name for event store details

Required	Yes
Default	IBM_EVENT
Property type	String
Usage	After clicking the advanced button on the connection properties window on the external service wizard, this property will display under the "Event configuration" tab. The two values listed are IBM_EVENT and IBM2. If you create a custom event component name, you can specify the value for it in the list box.
Globalized	Yes
Bidi supported	No

Siebel repository name (SiebelRepositoryName)

This property specifies the name of the Siebel repository from which the objects are to be discovered.

Table 78. Siebel repository name details

Required	Yes
Default	Siebel Repository
Property type	String
Usage	This default value is Siebel Repository. Although this is a required field, it is optional on the external service wizard. You can edit this value to point to other repositories if needed.
Globalized	No
Bidi supported	No

Siebel server view mode (SiebelServerViewMode)

This property specifies the Siebel Business Applications server view mode and controls the data that can be retrieved and what actions can be performed on it.

Table 79. Siebel server view mode details

Required	Yes
Default	3
Property type	Integer
Usage	This property displays after clicking the advanced button on the connection properties window on the external service wizard. This mode, when set to "Type of Siebel objects to discover" applies only to Siebel business objects, not to Siebel business services. The values supported by Siebel are 1 to 9.
Globalized	No

Type of Siebel objects to discover (TypeOfSiebelObjectsToDiscover)

This property specifies the type of Siebel object that needs to be discovered and listed.

Table 80. Type of Siebel objects to discover details

Required	Yes
Possible values	Siebel business objects or Siebel business services
Default	Siebel business objects
Property type	String
Usage	Although the default is Siebel business objects, you can select Siebel business services. Based on your selection, the external service wizard retrieves either the business objects or the business services.
Globalized	No
Bidi supported	No

Use resonate support for load balancing on Siebel server (UseResonateSupportForLoadBalancingOnSiebelServer)

This property indicates whether the Siebel Business Applications server uses resonate support.

Table 81. Use resonate support for load balancing on Siebel server details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	This property displays after clicking the advanced button on the connection properties window on the external service wizard. If the check box is selected, it will be set to True, and the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently. If the check box is not selected, it will be set to false.
Globalized	No

User name (UserName)

This property specifies the user name that is used to log in to the Siebel application.

Table 82. User name details

Required	Yes
Default	No default value
Property type	String
Usage	If a J2C Authentication Alias is used, then a user name is not required.
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 in 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 no longer required in version 7.0, but are supported for compatibility with previous versions:

- LogFileMaxSize
- LogFileName
- LogNumberOfFiles
- TraceFileMaxSize
- TraceFileName
- TraceNumberOfFiles

The following table lists and describes the resource adapter properties. A more detailed description of each property is provided in the sections that follow the table. For more information about how to read the property detail tables in the sections to follow, see “Guide to information about properties” on page 132.

Table 83. Resource adapter properties

Property name		Description
In the wizard	In the administrative console	
Adapter ID	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.
Event delimiter	EventDelimiter	Specifies whether the delimiter used between two name value pairs contains the object key name and value.
(Not available)	LogFileMaxSize	Deprecated
(Not available)	LogFilename	Deprecated
(Not available)	LogNumberOfFiles	Deprecated

Table 83. Resource adapter properties (continued)

Property name		Description
In the wizard	In the administrative console	
Resonate support	resonateSupport	Specifies that if resonate support is installed on the Siebel Business Applications server, and the value is set to True, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
Siebel server view mode	viewMode	Specifies the Siebel view mode and controls the type of data to be retrieved and the actions to be performed.
(Not available)	TraceFileMaxSize	Deprecated
(Not available)	TraceFileName	Deprecated
(Not available)	TraceNumberOfFiles	Deprecated

Adapter ID (AdapterID)

This property identifies a specific deployment or instance of the adapter.

Table 84. Adapter ID details

Required	Yes
Default	001
Property type	String
Usage	<p>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, SEBLRA, 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 SEBLRA001.</p> <p>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.</p> <p>For example, when you set the adapter ID property of two instances of WebSphere Adapter for Siebel Business Applications to 001 and 002. The component IDs for those instances, SEBLRA001 and SEBLRA002, 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 SEBLRAInstanc.</p> <p>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.</p>
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 85. 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

Enable high availability support (enableHASupport)

Do not change this property. It must be set to true.

Event delimiter (EventDelimiter)

This property indicates that the delimiter used between two name value pairs contains the object key name and value.

Table 86. Event delimiter details

Required	Yes
Default	;
Property type	String
Usage	If multiple value pairs are set against the object key in the event component, they will be used for the delimiter.
Globalized	No

Log file maximum size (LogFileMaxSize)

This property specifies the size of the log files in kilobytes.

Table 87. Log file maximum size details

Required	No
Default	0
Property type	Integer

Table 87. Log file maximum size details (continued)

Usage	When the log file reaches its maximum size, the adapter starts 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 88. 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 89. 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

Resonate support (ResonateSupport)

This property indicates whether the Siebel Business Applications server uses resonate support.

Table 90. Resonate support details

Required	No
Possible values	True False
Default	True
Property type	Boolean
Usage	If the value of this property is set to True, the adapter takes advantage of the load balancing feature to connect to the Siebel Business Applications server more efficiently.
Globalized	No

Siebel server view mode (SiebelServerViewMode)

This property specifies the Siebel view mode and controls the data that can be retrieved and the actions to be performed on it.

Table 91. View mode details

Required	Yes
Default	3
Property type	Integer
Usage	The View mode property applies only to Siebel business objects and not to Siebel business services.
Globalized	No

Trace file maximum size (TraceFileMaxSize)

This property specifies the size of the trace files in kilobytes.

Table 92. 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 93. 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 94. Trace number of files details

Required	No
Default	1
Property type	Integer

Table 94. 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 a message endpoint. You can set the activation specification properties either through the external service wizard or the administrative console.

Activation specification properties are used during the endpoint activation to notify the adapter of eligible event listeners. During inbound processing, the adapter uses these event listeners to receive the events before forwarding them to the endpoint.

The following table lists the activation specification properties for inbound communication. You set the activation specification properties in the external service wizard and can change them in the Integration Designer Assembly Editor, or after deployment through the administrative console. 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 132.

Table 95. Activation specification properties

Property name		Description
In the wizard	In the administrative console	
Connection URL	connectString	The connection URL to connect to the Siebel application.
“Code page (codePage)” on page 170	codePage	Specifies the code page that the adapter uses to connect to Siebel server.
Delivery type	DeliveryType	Determines the order in which events are delivered by the adapter to the export.
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.
Ensure once-only event delivery	AssuredOnceDelivery	Specifies whether the adapter provides assured once delivery of events.
Event component name	eventComponentName	The name of the Siebel component for the event table.
Event types to process	EventTypeFilter	A delimited list of event types that indicates to the adapter which events it should deliver. If the value of the property is blank or set to “*”, all event types are included. In addition, the property supports filtering at the operations level.
Include empty elements in the output	includeEmptyField	Specifies if the empty elements are included in the response business object.
Interval between polling periods	PollPeriod	The length of time that the adapter waits between polling periods.
Language code	languageCode	The language code used to log on to the Siebel Business Applications server.

Table 95. Activation specification properties (continued)

Property name		Description
In the wizard	In the administrative console	
Maximum connections	MaximumConnections	The maximum number of connections that the adapter can use for inbound event delivery.
Maximum events in polling period	PollQuantity	The number of events the adapter delivers to the export during each poll period.
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.
Password	password	Password for the corresponding user name.
Retry limit for failed events	FailedEventRetryLimit	The number of times the adapter attempts to redeliver an event before marking the event as failed.
Retry EIS connection on startup	RetryConnectionOnStartup	Controls whether the adapter retries the connection to the Siebel application 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.
User name	userName	The user name that is used to log in to the Siebel application.

Connection URL (ConnectionURL)

This property specifies the connection URL to connect to the Siebel application.

Table 96. Connection URL details

Required	Yes
Default	No default value
Property type	String
Usage	<p>Protocol://machinename:port/enterprisename/object manager/server name.</p> <ul style="list-style-type: none"> For Siebel 7.0.5 to 7.5x: siebel://<IP ADDRESS>/siebel/SSEObjMgr_ENU/sebldev1. For Siebel 7.8: siebel://<IP ADDRESS>:2321/Sieb78/SSEObjMgr_enu. For Siebel 8: siebel://<IP ADDRESS>:2321/SBA_80/SSEObjMgr_enu. The default port number is 2320. <p>In the examples provided (for Siebel versions 7.8 and 8), another port (2321) has been specified.</p>
Globalized	Yes
Bidi supported	Yes

Code page (codePage)

Specifies the code page that the adapter uses to connect to Siebel server.

Table 97. Code page details

Required	No
Default	No default value
Property type	String
Usage	<p>The Siebel Java Data Bean supports a limited list of code page settings. If the current code page setting of your run time is not supported, the adapter may not be able to connect to the Siebel server.</p> <p>This property allows the users to specify the code page details that the adapter uses to connect to Siebel server.</p> <p>Note: The adapter uses the code page property to change the “file.encoding” property of the current Java Virtual Machine (JVM), when it establishes its first connection to the Siebel server. This may affect other applications which run on the same JVM. You need to ensure that the code page has a valid value for all applications.</p>
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 98. Delivery type details

Required	No
Possible values	ORDERED UNORDERED
Default	ORDERED
Property type	String
Usage	<p>The following values are supported:</p> <ul style="list-style-type: none">• ORDERED: The adapter delivers events to the export one at a time.• UNORDERED: The adapter delivers all events to the export at once. <p>Note: If the value of DeliveryType is UNORDERED, then “Minimum connections (MinimumConnections)” on page 175 and “Maximum connections (MaximumConnections)” on page 174 must be greater than 1. Otherwise, event delivery will not work correctly. As an example, you can set Minimum connections as 2 and Maximum connections as 4.</p>
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 99. 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

Ensure once-only event delivery (assuredOnceDelivery)

This property specifies whether to provide ensure once-only event delivery for inbound events.

Table 100. 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 will be 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

Event component name (EventComponentName)

This property specifies the name of the event store where events are stored for inbound processing.

Table 101. Event component name details

Required	Yes
Default	<ul style="list-style-type: none"> • For Siebel version 7.x: IBM2 • for Siebel version 8 :IBM Event
Property type	String

Table 101. Event component name details (continued)

Usage	If you select one of the default values to configure the event business component on the Siebel Business Applications server, it is the name of the Siebel event business component that was created. You can also select a value from the list of values provided by the adapter. The list of values can be edited. If you create your own Siebel event business component, you can edit the list to include the name of that event business component.
Globalized	Yes
Bidi supported	Yes

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 102. 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	<p>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 indicates that no filter will be applied and that all events will be delivered to the export.</p> <p>Syntax: BOName:Operation1 Operation2, BOName:Operation1 Operation2 Operation3</p> <ul style="list-style-type: none"> • "," 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.
Example	<ul style="list-style-type: none"> • To receive all events of the Customer and Order business objects, specify one of these values: Customer:Create Update Delete,Order:Create Update Delete or Customer,Order. • 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. <p>Note: Only operations chosen for the business objects and services (see "Selecting business objects" on page 62) in the external service wizard can be used in this property. Otherwise, the events of those operations are not delivered properly during run time.</p> <p>If the EventTypeFilter and AdapterInstanceEventFilter 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 ConnectorId column matches the AdapterInstanceEventFilter property.</p>
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 103. Retry limit for failed events details

Required	No
Possible values	Integers
Default	5
Property type	Integer
Usage	<p>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:</p> <p>Default</p> <p>If this property is not set, the adapter tries five additional times before marking the event as failed.</p> <p>0</p> <p>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.</p> <p>> 0</p> <p>For integers greater than zero, the adapter retries the specified number of times before marking the event as failed.</p> <p>< 0</p> <p>For negative integers, the adapter does not retry failed events.</p>
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 104. Interval between polling periods details

Required	Yes
Possible values	Integers greater than or equal to 0.
Default	2000
Unit of measure	Milliseconds
Property type	Integer
Usage	<p>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 will occur immediately to make up for the lost time caused by the delay.</p>
Globalized	No
Bidi supported	No

Include empty elements in the output (includeEmptyField)

Specifies if the adapter returns the elements that have an empty string value in the response business object.

Table 105. Include empty elements in the output details

Required	No
----------	----

Table 105. Include empty elements in the output details (continued)

Possible values	True False
Default	False
Property type	Boolean
Usage	This property enables you to include the empty elements in the output. When this property is selected, the adapter returns the elements that have empty string values in the response business object.
Globalized	No
Bidi supported	No

Language code (LanguageCode)

This property specifies the language code used to log on to the Siebel Business Applications server.

Table 106. Language code details

Required	Yes
Default	ENU
Property type	String
Usage	If the system locale is English, the value for this property is ENU (English). This value is used to log on to the Siebel Business Applications 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 107. 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

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 108. Maximum events in polling period details

Required	Yes
Default	10
Property type	Integer

Table 108. Maximum events in polling period details (continued)

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

Minimum connections (MinimumConnections)

This property specifies the minimum number of connections that the adapter can use for inbound event delivery.

Table 109. 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

Number of times to retry the system connection (retryLimit)

This property specifies the number of times the adapter tries to reestablish an inbound connection.

Table 110. Number of times to retry the system connection details

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 Siebel application 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 Siebel application when it is first started, use the RetryConnectionOnStartup property.
Globalized	No
Bidi supported	No

Password (Password)

This property specifies the password for the corresponding user name.

Table 111. Password details

Required	Yes
Default	No default value
Property type	String

Table 111. Password details (continued)

Usage	This property appears after clicking Advanced in the connection properties window of the external service wizard. The password is saved in .import and .export files. Therefore, the adapter is able to connect to the Siebel application after it has been deployed. If a J2C Authentication Alias is used, then a password is not required.
Example	sadmin
Globalized	Yes
Bidi supported	Yes

Retry EIS connection on startup (retryConnectionOnStartup)

This property controls whether the adapter attempts to connect again to the Siebel application if it cannot connect at startup.

Table 112. Retry EIS connection on startup details

Required	No
Possible values	True False
Default	False
Property type	Boolean
Usage	<p>This property indicates whether the adapter should retry the connection to the Siebel application if the connection cannot be made when the adapter is started:</p> <ul style="list-style-type: none"> • Set the property to <code>False</code> when you want immediate feedback about whether the adapter can establish a connection to the Siebel application, 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 <code>Stopped</code>. After you resolve the connection problem, start the adapter manually. • Set the property to <code>True</code> 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 <code>RetryInterval</code> property to determine how frequently to retry and the value of the <code>RetryLimit</code> property to retry multiple times until that value is reached. The administrative console shows the application status as <code>Started</code>.
Globalized	No
Bidi supported	No

Retry interval if connection fails (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 113. Retry interval details

Required	Yes
Default	60000 (1 minute)
Unit of measure	Milliseconds
Property type	Integer

Table 113. Retry interval details (continued)

Usage	If this value is non-negative and the adapter encounters an error related to the inbound connection to the Siebel application, the adapter continues to attempt a connection after the specified interval until a connection is established.
Globalized	No
Bidi supported	No

Stop the adapter when an error is encountered while polling (StopPollingOnError)

This property specifies whether the adapter stops polling for events when it encounters an error during polling.

Table 114. 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

User name (UserName)

This property specifies the user name that is used to log in to the Siebel application.

Table 115. User name details

Required	Yes
Default	No default value
Property type	String
Usage	This property appears after clicking Advanced in the connection properties window of the and .export files. The user name authentication enhances the adapter to connect to the Siebel application after it is deployed. If a J2C Authentication Alias is used, then a password is not required.
Globalized	Yes
Bidi supported	Yes

Globalization

WebSphere Adapter for Siebel Business Applications 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 (Hebrew or Arabic, for example) and left-to-right (a URL or file path, for example) 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, and 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 encoding 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 or WebSphere Enterprise Service Bus uses 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, as well as properties that identify content or metadata to which transformation applies.

Bidirectional script data formats

IBM Business Process Manager or WebSphere Enterprise Service Bus use the bidirectional format of ILYNN (implicit, left-to-right, on, off, nominal). This is the format used by Windows. If an enterprise information system uses a different format, the adapter converts the format 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 116. Bidirectional format attributes

Letter position	Purpose	Values	Description	Default setting
1	Order schema	I	Implicit (Logical)	I
		V	Visual	

Table 116. Bidirectional format attributes (continued)

Letter position	Purpose	Values	Description	Default setting
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 by specifying values for each of the five bidirectional format attributes (listed in the preceding table). The `BiDiContextEIS` property can be set for the managed connection factory and the activation specification.

To identify event persistence data subject to transformation, set the `BiDiFormatEP` property by specifying values for each of the five bidirectional format attributes (listed in the preceding table). The `BiDiFormatEP` property can be set for the activation specification.

To identify application-specific data for transformation, annotate the `BiDiContextEIS` property and the `BiDiMetadata` property within a business object. Annotate the properties by using the business object editor within IBM Integration Designer to add the properties as application-specific elements of a business object.

Related reference

“Activation specification properties” on page 168

Activation specification properties hold the inbound event processing configuration information for a message endpoint. You can set the activation specification properties either through the external service wizard or the administrative console.

“Managed connection factory properties” on page 145

Managed connection factory properties are used by the adapter at run time to create an outbound connection instance with the Siebel application.

Properties enabled for bidirectional data transformation

Bidirectional data transformation properties enforce the correct format of bidirectional script data exchanged between an application or file system and integration tools and runtime environments. After 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 used in the external service wizard

The following connection properties for the external service wizard are enabled for bidirectional script data transformation:

- Connect string
- Password
- Prefix
- User name

Managed connection properties

The following managed connection properties are enabled for bidirectional script data transformation:

- Connect string
- Password
- User name

Activation specification properties

The following activation specification properties are enabled for bidirectional script data transformation:

- Connect string
- Password
- User name

Adapter messages

View the messages issued by WebSphere Adapter for Siebel Business Applications at the following location.

Link to messages: http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/topic/com.ibm.wbpm.ref.doc/topics/welc_ref_msg_wbpm.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 CWYEB are issued by WebSphere Adapter for Siebel Business Applications
- 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 Siebel Business Applications.

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://www14.software.ibm.com/webapp/wsbroker/redirect?version=pix&product=wps-dist&topic=bpmroadmaps>.

- 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, 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>.
- IBM Business Process Manager, version 7.0, 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/v7r0mx/index.jsp>.
- WebSphere Adapters, version 6.2.x, information center: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/index.jsp>.
- IBM WebSphere Adapters, version 7.5 installation on WebSphere Application Server, version 8.0 information: <http://www-01.ibm.com/support/docview.wss?rs=695&uid=swg27011040>.

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 Adapter video demos 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 will open 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 demos 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
- Lotus® Domino®: Configuring notification rules for forms in database paths, and generating their events
- IBM i: Supporting CCSID in the RPG, COBOL, and service programs
- IBM i: Discovering the PCMLS in a local file system

- JDE: Retrieving data using multiple business functions in outbound processing
- Oracle: Integrating support for an Oracle AQ activity
- Oracle: Integrating an XML gateway through JMS in Oracle E-Business Suite
- PeopleSoft: Handling a RecordNotFound fault behavior
- PeopleSoft: Handling an event filtering
- SAP: Monitoring Tivoli ITCAM for a BPM Portfolio requirement
- SAP: Adding random numbers to the end of duplicate field names
- Siebel: Adding codepage properties to both EMD and runtime for an inbound Business Object scenario
- Siebel: Supporting more than one output argument for a Siebel Business Services Interface in an outbound scenario
- WAT: Creating a custom adapter using WebSphere Adapter Toolkit
- WAT: Creating a custom adapter outbound interface in WebSphere Adapter Toolkit
- WAT: Testing a module in WebSphere Adapter Toolkit

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](http://www.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

A

- activation specification properties
 - reference
 - inbound configuration 168
 - setting in administrative console 99, 104
 - setting with external service wizard 168
- Active-Passive 24
- adapter 39
 - architecture 3
 - configure 39
 - implementation 17
 - information resources 180
 - messages 180
 - package files 113
 - performance 106
 - project, create 45
 - related information 180
 - samples and tutorials 180
 - support and assistance 180
 - technotes 180
- adapter application
 - starting 105
 - stopping 106
- Adapter for Siebel Business Applications module
 - exporting as EAR file 91
 - installing EAR file on server 92
 - starting 105
 - stopping 106
- AdapterStyle 133, 156
- application-specific information
 - business-object-level metadata 125
 - container attribute-level metadata 125
 - property-level metadata 125
 - siebel business components 125
 - siebel business objects 125
 - siebel business services 125
- ApplyChanges operation 4
- artifacts 33
- AssuredOnceDelivery 168
- authentication 39
 - description 18
 - in the wizard 18
 - runtime environment 18
- authentication alias 39
 - creating 43
 - J2C 18

B

- Batch Processing 24
- bidirectional data transformation
 - properties
 - activation specification properties 180
 - connection properties
 - external service wizard 180

- bidirectional data transformation
 - properties (*continued*)
 - managed connection properties 180
- business faults
 - DuplicateRecordFault 131
 - MatchesExceededLimitFault 131
 - MissingDataFault 131
 - overview 15
 - RecordNotFoundFault 131
 - WBIFault 15
- business graph 4, 12
- business integration adapters to JCA-compliant adapters 30
- business object 39
- business object information 125
- business objects 12
 - application-specific information 125
 - configuring for inbound processing 67
 - configuring for outbound processing 54
 - creating structure 12
 - naming conventions 128
 - selecting for inbound processing 62
 - selecting for outbound processing 49
 - supported data operations 127
- business services
 - selecting for inbound processing 65
 - selecting for outbound processing 52
- business-object-level metadata 125
- BusinessObjectNameSpace 133, 156

C

- CEI (Common Event Infrastructure) 110
- changes after migration
 - to the export file 35
 - to the import file 35
 - to the wsdl file 35
- CL (control language) 39
- clustered environment
 - adapters version conflict 24
 - deployment 24
 - inbound process 24
 - inbound processes 24
 - load balancing 24
 - outbound process 24
 - outbound processes 25
- CodePage 133, 156, 168
- Common Event Infrastructure (CEI) 110
- compatibility matrix 2
- confidential data, disguising 18
- confidential tracing 18
- configuration overview 40
- configuration properties
 - inbound 154
- configuring
 - logging properties 113
 - Performance Monitoring Infrastructure (PMI) 107

- connection properties
 - setting in the external service wizard 47
- connectionRetryInterval 145
- connectionRetryLimit 145
- connectionURL 133, 145, 156, 168
- connector project 45
- container attribute-level metadata 125, 126
- Create operation 5
- custom properties
 - activation specification 99, 104
 - managed connection factory 97, 102
 - resource adapter 95, 101

D

- data operations 127
- debugging
 - self-help resources 123
- Delete operation 6
- DelimiterForKeysInTheEventStore 133, 156
- DeliveryType 168
- deployment 92
 - production environment 89
 - test environment 85
- deployment environment 39, 85
- deployment options 20
- developerWorks 181
- disguising confidential data 18

E

- EAR file
 - exporting 91
 - installing on server 92
- edit binding
 - import component, inbound 80
 - import component, outbound 79
- embedded adapter
 - activation specification properties, setting 99
 - considerations for using 21
 - managed connection factory properties, setting 97
 - resource adapter properties, setting 95
 - usage considerations 20
- embedded adapters
 - changing configuration properties 95
 - setting activation specification properties 99
 - setting managed (J2C) connection factory properties 97
 - setting resource adapter properties 95
- embedded deployment 86
- enableHASupport property 24
- errorOnEmptyResultSet 145

- event business components 9
- event delivery 170
- event store
 - event business components 9
 - event notification 3
 - event record 9
 - event triggers 9
 - polling 9
- event table
 - creating 41
- eventComponentName 168
- EventDelimiter 140, 163
- EventTypeFilter 168
- Exists operation 7
- export file 39
- exporting module as EAR file 91
- external dependencies, adding 85, 86
- external service wizard 12
- external software dependencies 45

F

- FFDC (first-failure data capture) 116
- files
 - SystemOut.log log file 115
 - trace.log trace file 115
- FilterFutureEvents 168
- first-failure data capture (FFDC) 116
- Folder 133, 156

G

- GenerateBusinessObjectsWithShorterNames 156
- globalization and bidirectional data transformation
 - properties t 178
 - script data formats 178
 - script data transformation 178

H

- HA Active-Active 24
- hardware and software requirements 2
- hardware requirements 2
- High Availability (HA)
 - clustered environments 180
- high-availability environment 24
 - Active-Active 24
 - Active-Passive 24
 - deployment 24
 - inbound processes 24
 - outbound processes 25

I

- IBM Business Process Manager
 - information 181
- IBM Business Process Manager or WebSphere Enterprise Service Bus
 - deploying to 89
- IBM Business Process Manager, version 7.0, information 181
- IBM Integration Designer
 - information 181

- IBM Integration Designer (*continued*)
 - test environment 85
- IBM WebSphere Adapter for Siebel Business Applications
 - administering 95
- IBM WebSphere Adapter Toolkit 181
 - developerWorks resources 180
- IBM WebSphere Enterprise Service Bus
 - information 181
- implementation, Java 87
- import component, inbound 80
- import component, outbound 79
- import file 39
- inbound
 - configuration properties 154
- inbound configuration
 - reference
 - activation specification
 - properties 168
- inbound processing 39
 - business graph 8
 - configure the module
 - selecting business objects 62
 - selecting business services 65
 - generating the service 69
 - introducing 8
 - setting deployment properties 69
- includeEmptyField 145, 168
- installing EAR file 92
- interaction specification properties 153
 - changing 83
- interface 39
- IOWrapper
 - 133, configure 52
 - reference 125
- iterative development
 - inbound
 - connection-based editing 80
 - Edit Binding 80
 - import component 80
 - outbound
 - connection-based editing 79
 - Edit Binding 79
 - import component 79

J

- JAR file 45
- JAR file, adding external 85, 86
- Java 2 security 18
- Java implementation 87

L

- LanguageCode 133, 145, 156, 168
- load balancing 24
- Log Analyzer 113
- log and trace
 - configure 113
- Log and Trace Analyzer, support for 14
- log and trace files 14
- log files
 - changing file name 115
 - disabling 113
 - enabling 113
 - level of detail 113

- log files (*continued*)
 - location 115
 - SystemOut.log 115
- logging
 - configuring properties with
 - administrative console 113
 - logging level 113

M

- managed (J2C) connection factory
 - properties
 - reference
 - embedded adapters 145
 - setting in administrative console 97, 102
- matrix, compatibility 2
- MaximumConnections 168
- MaximumRecords 153
- messages, adapter 180
- metadata
 - attribute level 125
 - attribute-level 126
 - business object level 125
 - business-object-level 125
 - property level 125
 - property-level 126
- MethodName 133, 156
- migration 30
 - adapter-specific artifacts 33
 - artifacts
 - adapter-specific 33
 - considerations
 - compatibility with earlier versions 25
 - migrating applications from WebSphere InterChange Server
 - roadmap 31
 - overview 31
 - performing migration 27
 - upgrading a project 29
 - WebSphere InterChange Server migration
 - migration roadmap 31
 - WebSphere InterChange Server migration wizard 32
- MinimumConnections 168
- module 39
 - adding to the server 88
 - configuring for deployment
 - overview 39
 - configuring inbound processing 62
 - configuring outbound processing 49
 - deploy for testing 85
 - monitoring performance 106
 - multiple connection 170

N

- naming conventions for business objects
 - business graph 128
 - siebel business components 128
 - siebel business services 128
 - siebel integration components 128

O

- outbound configuration
 - reference
 - connection properties for the wizard 133, 156
 - interaction specification properties 153
 - managed connection factory properties 145
 - resource adapter properties 140, 163
- outbound configuration properties 132
- outbound operations 4
 - ApplyChanges 4
 - Create 5
 - Delete 6
 - Exists 7
 - Retrieve 7
 - RetrieveAll 7
 - Update 8
- outbound processing 39
 - configure the module
 - selecting business objects 49
 - Configure the module
 - selecting business services 52
 - generating the service 56
 - outbound process 56
 - setting deployment properties 56
 - supported outbound operations
 - business object hierarchy 3
 - testing the module 88

P

- package files for adapters 114
- Password 133, 145, 156, 168
- performance monitoring
 - infrastructure 106
 - configuring 107
 - performance statistics 109
- Performance Monitoring Infrastructure (PMI)
 - configuring 107
 - description 106
 - viewing performance statistics 109
- performance statistics 109
- PMI (Performance Monitoring Infrastructure)
 - configuring 107
 - viewing performance statistics 109
- PollPeriod 168
- PollQuantity 168
- Prefix 145
- PrefixForBusinessObjectName 133, 156
- problem determination
 - self-help resources 123
 - solutions to common problems 116
- project 39
- project interchange (PI) file
 - project interchange files 29
 - projects 29
 - updating without migrating 29
- properties
 - activation specification 99, 104
 - configuration properties
 - inbound 154

- properties (*continued*)
 - configuration properties (*continued*)
 - outbound 132
 - inbound configuration 154
 - managed (J2C) connection factory 97, 102
 - outbound configuration 132
 - resource adapter 95, 101
- properties information guide 132, 154
- property-level metadata 125, 126

R

- RAR (resource adapter archive) file
 - description 90
 - installing on server 90
- recommended fixes 123
- Redbooks, IBM WebSphere Adapters 180
- related products, information 180
- requirements
 - hardware 2
 - software 2
- ResonateSupport 140, 145, 163
- resource adapter archive (RAR) file
 - description 90
 - installing on server 90
- resource adapter properties 140, 163
 - setting in administrative console 95, 101
- Retrieve operation 7
- RetrieveAll operation 7
- Retry limit property 175
- RetryInterval 168
- RetryLimit 168
- roadmap
 - configuring the module 39
- roadmap for migrating WebSphere InterChange Server applications 31
- runtime environment
 - deploying EAR file 89

S

- samples 37, 181
- security
 - disguising sensitive data 18
 - user authentication 18
- security, Java 2 18
- self-help resources 123
- sensitive data, disguising 18
- service 3
- service interface queue 41
- SiebelBusinessObjectNameForEventStore 133, 156
- SiebelRepositoryName 133, 156
- SiebelServerViewMod 133, 156
- SiebelServerViewMode 140, 163
- software dependencies, adding
 - external 45, 85, 86
- software requirements 2
- solutions to common problems
 - adapter does not generate an exception 116

- solutions to common problems (*continued*)
 - adapter does not generate artifacts properly 116
 - adapter generates multiple container attributes 116
 - adapter may time out 116
- stand-alone adapter 104
 - considerations for using 22
 - managed connection factory
 - properties, setting 102
 - resource adapter properties, setting 101
 - usage considerations 20
- stand-alone adapters
 - changing configuration
 - properties 101
 - setting activation specification
 - properties 104
 - setting managed (J2C) connection factory properties 102
 - setting resource adapter
 - properties 101
- starting adapter applications 105
- stopping adapter applications 106
- stopPollingOnError 168
- StopPollingOnError 168
- support
 - overview 113
 - plug-in for IBM support assistant 123
 - self-help resources 123
 - web site 123
- supported data operations 127
- SystemOut.log file 115

T

- target component 87
- technotes 2, 123
- test environment 85
 - adding module to 88
 - deploying to 88
 - testing modules 88
- trace files
 - changing file name 115
 - disabling 113
 - enabling 113
 - level of detail 113
 - location 115
 - trace.log 115
- tracing
 - configuring properties with administrative console 113
- triggers, event 9
- troubleshooting
 - overview 113
 - self-help resources 123
- tutorials 37, 181
- TypeOfSiebelObjectsToDiscover 133, 156

U

- UNORDERED 170
- Update operation 8

UseResonateSupportForLoadBalancingOnSiebelServer 133,
156
UserName 133, 145, 156, 168

V

ViewMode 145

W

WebSphere adapter for Siebel Business
Applications
 Inbound processing 8
 Outbound processing 3
 planning 17
 technical overview 1
WebSphere Adapters, version 6.0,
 information 181
WebSphere Adapters, version 6.2.x,
 information 181
WebSphere Application Server
 information 181
WebSphere business integration
 adapters 30
WebSphere Business Integration Adapters
 information 181
WebSphere Extended Deployment 24
wiring components 87



Printed in USA